Classification Of Lymph Node Metastases In Breast Cancer With Features From Tissue Images Using Machine Learning Techniques

Master's thesis
Master's degree programme in Bioinformatics
Faculty of Medicine and Life Sciences
University of Tampere
Jyoti Prasad Bartaula
May 2017

Acknowledgement

This thesis work was carried out in partial fulfillment of the requirement for Master's degree programme in Bioinformatics, at Faculty of Medicine and Life Sciences, University of Tampere.

Foremost, I would like to extend my sincere gratitude to my supervisor Thomas Liuksiala for his continuous support throughout this thesis work. His constant guidance and motivations were instumental for completion of this project work. I am indebted to university researcher Dr. Pekka Ruusuvuori for providing me with this research topic and data needed for completion of it, and also for valuable suggestions throughout the research work and for comments on final manuscript. I would like to express my deepest appreciation to prof. Matti Nykter for reviewing this manuscript and helping me to sort out other practical problems that were there during research work. I am also grateful to university lecturer Juha Kesseli for his help to plan my study during entire study period. I would also like to thank my friends and colleagues for their everlasting help and inspiration throughout the process. Last but not least, I express my profound gratitude to my parents for providing me with boundless support and blessing throughout my life.

May 2017

Jyoti Prasad Bartaula

Abstract

UNIVERSITY OF TAMPERE
Master's Degree Programme in Bioinformatics
JYOTI PRASAD BARTAULA: Classification Of Lymph Node Metastases In Breast Cancer With Features From Tissue Images Using Machine Learning Techniques
Master of Science Thesis, 64 pages, 2 Appendix pages
May 2017
Major subject: Bioinformatics
Supervisor: Thomas Liuksiala
Reviewers: Professor Matti Nykter, Juha Kesseli
Keywords: Lymph node metastasis, Breast cancer, Feature selection, Classification

Determining the metastatic involvement of lymph node is very crucial in designing the treatment plans in breast cancer. Traditional way of detecting the lymph node metastases involves manual histopathological examination of specimen, which is subjective and tiresome process. In this thesis, an automated system to classify lymph node metastases in breast cancer with features from digitized tissue images is proposed. The proposed system consists of applying different machine learning algorithms for classification together with various feature selection techniques.

minimum Redundancy Maximum Relevance(mRMR), wrapper methods, area under the ROC curve of random forest (AUCRF), and least absolute shrinkage and selection operator (LASSO) were implemented to select the most relevant features among 214 original features. Various classification models were learned using selected features to classify between metastatic and non-metastatic samples. Among the models learned, random forest model showed to perform better than others.

The results obtained from this thesis show encouraging signs for automated classification of lymph node metastases in breast cancer with features from digitized tissues images with the application of machine learning techniques. Also, results show that feature selection helps in removing irrelevant and redundant features, which not only deceases the computational time of classification algorithms but can also enhances the classification performance.

Contents

Acronyms

ANN	Artificial neural network
AUC	Area under the curve
AUCRF	Area under the ROC curve of random forest
BFS	Best-first search
CAD	Computer-assisted diagnosis
KNN	K-nearest neighbor
LASSO	Least absolute shrinkage and selection operator
MLP	Multi-layer perceptrons
mRMR	minimum Redundancy Maximum Relevance
OOB	Out-Of-Bag
PCA	Principal component analysis
PCs	Principal components
RF	Random forest
ROC	Receiver operator characteristic
SVM	Support vector machine

1. Introduction

One of the hallmarks (traits) of cancerous cells is their ability to invade surrounding tissues and thence travel to other body parts (Hanahan and Weinberg, 2000). Cancer which has spread to other body part is known as metastatic cancer, and the process is know as *metastasis*. Cancer cells can travel from the part of body where they originate (known as primary site) to other part of body either through blood stream or the lymph system. If cancerous cell travels through the lymphatic system, lymph node/s where cancer cells are likely to reach first are called *sentinel nodes*. And it is found that the lymph node nearer to primary site is often the first site of metastases (Sleeman and Thiele, 2009; Alitalo and Carmeliet, 2002). In accordance with this finding, the lymph nodes located under the arm, called *axillary lymph nodes*, are the first place where breast cancer is likely to spread. In other words, in case of breast cancer, sentinel node/s are mostly likely to be within axillary lymph nodes. Determining the metastatic involvement of the axillary node is one of the most significant prognostic variables in breast cancer (Tafreshi et al., 2012). If lymph node status is negative, cancer is more curable and patients have higher chance of surviving in contrast to lymph node-positive breast cancer which has poorer prognosis. Therefore, detecting the status of the lymph node is important in designing the treatment plan and elucidating the progression of breast cancer.

Traditional way of detecting the lymph node metastasis is based on histopathological examination. This classical way of manual inspection of histopathological images by pathologist is tedious and sluggish. Furthermore, manual inspection of samples is very much visual and subjective process even though it requires skills, experiences, and is based on educated opinions of pathologists (Gurcan et al., 2009; Sertel, 2010). Also, this subjective analysis suffers from inter and intra observer variations in diagnostic outcome despite putting effort into standardizing the process(Metter et al., 1985; Al-Kofahi et al., 2011; Stenkvist et al., 1983). And such variations might result in making inappropriate decision and inaccurate predictions of clinical outcomes with serious clinical consequences(Sertel, 2010). Furthermore, in the special case of metastasis detection, other factors like variable staining and incomplete section screening might lead to missing metastases in classical way of screening (Weaver et al., 2003).

With the advent of creating the digital images of glass slides together with increase in computer power and development of various image analysis techniques, it is now possible to implement the computer-assisted approaches in analyzing digital histopathological images incorporating various digital image analysis and pattern recognition techniques (Kårsnäs, 2014; Gurcan et al., 2009; Sertel, 2010). There are manifold advantages of implementing computer-assisted approaches, otherwise knows as computer-assisted diagnosis in digital histopathology. Computer-assisted diagnosis (CAD) application in histopathological image analysis brings quantitative interpretation of digital slides. The quantitative histopathology in turn brings objectivity in assessing prognosis, with improved diagnostic and prognostic capabilities and increase in the accuracy in predicting the clinical outcome (Sertel, 2010). Furthermore, quantitative approach of histopathology is important not only in clinical setting, but also of great significance in research fields, like to understand the underlying biological mechanism of disease development(Sertel, 2010; Gurcan et al., 2009). In addition, since histopathological examination is based on observation of tissue structure at various scales, computerized image analysis technique helps to extract more quantitative features providing more objective prognostic clues, which may not be easily observed by qualitative visual inspection (Sertel, 2010). To sum up, CAD application in histopathological image analysis helps in alleviating the problem that arises due to qualitative and subjective visual examination by pathologists and at the same time also reduces the work load of pathologists.

In this thesis work, an automated system for detection of lymph node metastases in breast cancer using features extracted from digitized tissue images is proposed. Many features are extracted from tissue images, but not all of them contribute in detection of metastasis. Thus, to find the relevant features, different feature selection techniques are studied. Finally, a system is developed implementing various machine learning algorithms using the selected features. The system will then tries to classify the lymph node tissue section either as metastatic region or non-metastatic region.

The structure of thesis is as described below. Chapter 2 introduces about pattern recognition and classification, and about dimensionality reduction with focus on feature selection techniques. Aims and objectives of this thesis work are laid out in chapter 3. Chapter 4 consists of description about material and methods used. Similarly, results and discussion are in chapter 5. Finally, conclusions drawn from the thesis work are presented in chapter 6.

2. Background

2.1 Pattern recognition and classification

Pattern recognition is concerned with the automatic discovery of regularities in data through the use of computer algorithms and with the use of regularities to take actions like classifying the data into different categories (Bishop, 2006). It is a branch of machine learning, and Samuel Arther defined machine learning as a "field of study that gives computers the ability to learn without being explicitly programmed" (Samuel, 1959). Hence, due to its ability to learn, machine learning techniques can be applied for automated pattern recognition in data when the patterns in data are beyond the human comprehension (Bishop, 2006). Pattern recognition can be categorized into two groups namely supervised pattern recognition and unsupervised pattern recognition. In supervised pattern recognition, also called as supervised learning, the aim is to assign patterns (also called as instances, observations, samples, or examples) to one of the predefined categories (class labels), whereas in unsupervised pattern recognition, also known as unsupervised learning, there are no class labels, and the aim is to partition the patterns into group or cluster based on hidden data regularities.

Depending upon the types of classes that are assigned to the patterns, supervised learning can be further divided into two groups: classification and regression. In classification, class labels are represented by finite number of discrete categories, whereas in regression, labels are one or more continuous variables. If there are only two categories of class labels, classification task is said binary classification, and if more than two categories of classes, it is called multiclass classification. Since this project work focuses on classification (binary) task, following section provides brief introduction to it.

Classification problem concerns with training a classifier or learning a model such that it can assign the class label to new patterns as accurately as possible. This training of the classifier is done by providing a set of input patterns called as *training set* to classification algorithm, where the output class labels for each patterns are known beforehand. The classifier thus learned is

then used to classify the new patterns to one of the predefined categories. The set of these new patterns that are not used to train the classifier is known as *test set*. The ability of model how accurately it can assign correct categories to patterns in test set is known as *generalization*. If the model can accurately assign class label to test set patterns, it is said to have higher generalization ability, in return, it is said to have higher predictive accuracy or lower prediction error.

In practice, classification task aims to learn a model that has as low prediction error as possible. And prediction error of the model can decomposed into two types: error due to bias and error due to variance (Witten and Frank, 2011; Hastie et al., 2009). According to conceptual definition from (Fortmann-Roe, 2012), the error due to bias is measure of how far is expected predictions of model from true value. Similarly, Fortmann-Roe defines the error due to variance as variability of model prediction for a given data point. When the model prediction is repeated multiple times, the variance measures how much model predicted values differ in each iterations. So, in this sense, the variance does not measure whether predictions are accurate or not, rather it measures how consistent are model predictions for a given data point between different realization of the model (Fortmann-Roe, 2012).

During model building process, it is highly desirable to realize a model with reduced bias and variance as much as possible. However, there is always a trade off between model's ability to minimize bias and variance. This problem of simultaneously minimizing the bias and variance is known as *bias-variance tradeoff*. If a model exhibits low variance but higher bias, the model is said to "underfitting" the data. Underfitting occurs when a model does not fit the data well. In other words, underfitting is the result of having too simple model such that it is unable to grasp the underlying trend of data. In opposite, a model is said to "overfitting" the data if it fits data too well, i.e., it even captures the noise in the data. Explaining in terms of the bias-variance trade off, overfitting model exhibits low bias but high variance. Both underfitting and overfitting lead to poor prediction accuracy. Hence, it is necessary to have a model which would be good fit to the data, and at the same time, also generalizes well to unseen data in order to have good prediction accuracy. In later section we will introduce some methods to estimate the prediction accuracy of the models.

Classification problem is a multi step task. The general steps involved in classification problem are data collection and representation, feature selection and/or feature reduction and actual classification. Data collection in classification task is mostly a problem specific, where data can

numeric, boolean or nominal values. Data, thus collected is represented into matrix form as the input to the classification algorithms. Generally, rows of matrix are the patterns or observations and columns are the features or variables. Features are the measurable quantity that describes an observation. So, each pattern is represented by a feature vector such that $F = (f_1, f_2, f_3, ...f_n)$. Next step of feature selection or feature reduction aims to reduce the number of features such that only those features that help to discriminate between the classes are retained. In many classification task, this step may or may not be needed. Finally, in classification step, a model is learned which would generalize to unseen data.

2.1.1 Classification algorithms

Classification algorithms form the backbone for classification task. Classification algorithms are trained with train set with the aim such that they generalizes well to test set. In following section, we will discuss about some popular classification algorithms within machine learning.

a) Instance based learning

In instance based learning, no model is learned for classification. Instead, stored training instances themselves represent as knowledge for classification. One of the instance based learning classifier is nearest neighbor classifier. It is one of the simple and effective classifier where distance metric is used to find closest instance in training set to each test instance (Witten and Frank, 2011). Once the closest instance (neighbor) in training set is found for a test instance, the class belonging to it is assigned to test instance. Commonly used distance metric is the Euclidean distance. This process of finding the closest neighbor from training set to test instances gives rise to rule known as nearest neighbor (NN) rule.

K-nearest neighbor (KNN) is one of the most popular algorithm based on nearest neighbor rule. Instead of looking for single nearest neighbor, KNN searches for k numbers of samples (neighbors) in training set that are nearest to test instance. And during classification, test instance is assigned the class label which is represented most among k neighbors. An example of KNN classification problem is shown in figure 2.1. From figure, green circle represents the test instance, which should be classified either to the first class represented by red triangles or to the second class represented by blue squares. If the number of neighbors (k) is chosen equal to be 3, test instance is assigned with the first class, i.e, red triangle, because among 3 neighbors, 2 of

them are red squares, but if k is chosen to be 5, test instance is assigned blue square as majority of squares among 5 neighbors are blue. From this discussion, it should be clear by now that the classification performance of KNN classifier depends upon number of neighbors (k) which is different from one data set to other, thus requiring the tuning of this parameter to find the optimal value of it. However, the rule of thumb is that k = $\sqrt{n}$ (Jirina and Jr, 2011) where n is size of training set. In addition, in case of binary classification (number of classes =2), usually only odd value of k is selected in order to avoid the tie situations. Tie situations arise when both classes receive same number of votes among k neighbors. Even though KNN is simple and easy to implement classifier, its disadvantages lies in the fact that KNN requires larger computational cost for testing as all the training instances are kept in memory (thus, requires large storage as well), and it is very sensitive to irrelevant and redundant features (Cunningham and Delany, 2007).

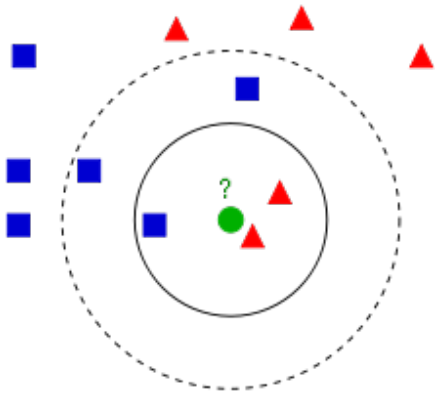


Figure 2.1: KNN- classification [1]

Another more sophisticated instance based learning algorithm is support vector machine (SVM). The original idea of SVM algorithm (Cortes and Vapnik, 1995; Vapnik, 1995) is to find the hyperplane that best divides the data points into two classes. There may be several of such hyperplanes that divide the data points into two classes, but the aim of SVM is to find the optimal hyperplane that should run between two classes where all data points should lie on one side of hyperplane in which they belong to such that hyperplane lies as far as possible from nearest data points from both classes.

Brief mathematical introduction to SVM is presented as below. Suppose there are L training data points represented by $\{x_i,y_i\}$, i=1,...L, and belong to either one of two class such that y_i =

[1] https://upload.wikimedia.org/wikipedia/commons/thumb/e/e7/KnnClassification.svg/220px-KnnClassification.svg.png

{+1,-1}, the hyperplane can be defined by equation:

$$\mathbf{w}.x + b = 0 \tag{2.1}$$

where **w** is normal to hyperplane, b is bias, x is point lying on hyperplane. Data points are said to be linearly separable if there exists a decision rule or a separating hyperplane for two classes fulfilling the inequalities: $\mathbf{w}.x_i + b \geq +1$ for y_i= +1 and $\mathbf{w}.x_i + b \leq -1$ for y_i = -1 (Cortes and Vapnik, 1995). And, now combining these two equations into:

$$y_i.(\mathbf{w}.x_i + b) - 1 \geq 0 \quad \forall i \tag{2.2}$$

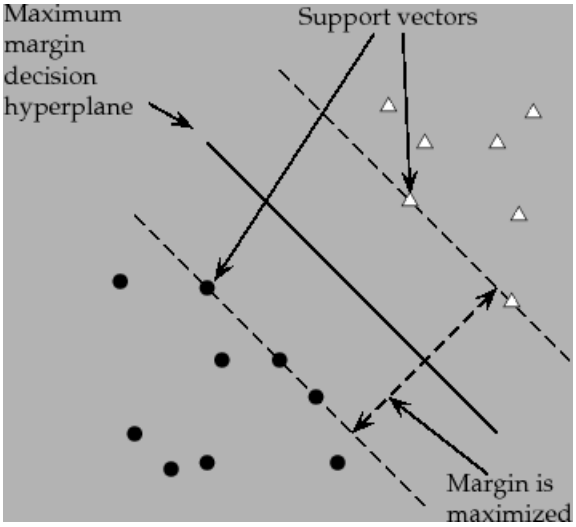


Figure 2.2: SVM hyperplane in linearly separable case[2]

The training samples thats falls on these two hyperplanes are called *support vectors* and can be described by:

$$x_i.\mathbf{w} + b = +1 \tag{2.3}$$

$$x_i.\mathbf{w} + b = -1 \tag{2.4}$$

These planes run parallel to optimal hyperplane and the distance between these planes and

[2]http://nlp.stanford.edu/IR-book/html/htmledition/img1260.png

optimal hyperplane is called margin. Since the aim of svm is to orient the hyperplane as far as possible from support vector, the maximization of margin is what it is needed. Example of hyperplane is shown in figure 2.2. The margin can be explained as $\frac{1}{\|\mathbf{w}\|}$ and maximizing it is equivalent to finding min $\frac{1}{2}\|\mathbf{w}\|^2$, which is quadratic optimization problem under the constraint of equation 2.2. The above described Support vector machine (SVM) is called as hard margin SVM, where all the training samples are correctly classified. However, with many real world data, it not possible to classify all the samples correctly, for example, when there is noise in data. Hence, in such case SVM is allowed to has minimal errors by introducing the variable known as *slack variables* $\xi_i \geq 0$; for all training samples. Now, with the introduction of slack variables, the constraint becomes (Cortes and Vapnik, 1995)

$$y_i.(\mathbf{w}.x_i + b) \geq 1 - \xi_i \quad \forall i \tag{2.5}$$

With this, the problem to be optimized becomes minimizing (Cortes and Vapnik, 1995)

$$\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{L}\xi_i \tag{2.6}$$

subject to constraint 2.5. The parameter C, which is called as cost parameter, is a constant value. It is a regularization parameter that controls how much we want to avoid misclassifying the training samples. In other words, parameter "C" is associated with finding the trade off between maximization of width of margin and minimizing the number of misclassified observations in training set. Above formulation of SVM is now called as soft margin SVM.

The above mentioned approach for SVM works only if decision boundary is in input space. In order to have decision boundary in feature space, a solution to non-linearly separable case, feature vectors are mapped to higher dimension in which the problem is most likely to become linearly separable. SVM does this non-linear transformation of input space to feature space with a technique called as *Kernel Trick*. If we recall, linear SVM relies on dot product between two vectors $K(x_i, x_j) = x_i^T x_j$, which means only dot product of mapped input in feature space needs to be calculated without requiring the explicit calculation of mapping function (Müller et al., 2001). This enables to use so called "trick", which is to replace dot product with kernel function.

The kernel function denotes a dot product in feature space, and is denoted as $K(x, y) = \Phi(x).\Phi(y)$, where Φ is a function that maps an instance into a feature space(Witten and Frank, 2011). There are many kernel function available but the most popular kernel used in SVM are (Hastie et al., 2009):

1. polynomial: $K(x_i, x_j) = (\gamma x_i^T x_j + \gamma)^d, \gamma > 0$
2. radial basis function (RBF): $K(x_i, x_j) = exp(-\gamma \|x_i - x_j\|^2), \gamma > 0$
3. sigmoid: $K(x_i, x_j) = tanh(\gamma x_i^T x_j + r)$

b) Bayes classifier

näive Bayes is a probalistic classifier based on applying Bayes' rule. Given a instance $\mathbf{X} = (x_1, x_2, ..., x_n)$ and class variable C_i, Bayes' rule can be stated as:

$$p(C_i|\mathbf{X}) = \frac{p(\mathbf{X}|C_i)p(C_i)}{p(\mathbf{X})}$$

where $p(C_i|\mathbf{X})$ is the probability of observing specific class C_i given instance $\mathbf{X}$. This probability is known as *posterior probability*. Similarly, $p(\mathbf{X}|C_i)$ is the probability of generating instance $\mathbf{X}$ given class C_i. The term $p(C_i)$ at the numerator known as *prior probability* is the probability of occurrence of that specific class without knowing the instance $\mathbf{X}$, and it can be calculated by observing frequencies of classes in training set. So, in plain word, Bayes rule can be written as:

$$\text{posterior probability} = \frac{\text{conditional probability} \quad . \quad \text{prior probability}}{\text{evidence}}$$

Now, näive Bayes classifier performs the classification by selecting the class that has maximum posterior probability. To estimate the probabilities, training data set is used. There are many ways to estimate the posterior probability. Maximum likelihood and Bayesian are the popular methods among them. naive Bayes inherits its adjective "näive" due to its assumption of class-conditional independence. This is it to say that naive Bayes classifier assumes that the value of one feature for given class is not affected by values of other features for given class. This assumption is made to simplify the computation. However, in real world this assumption does not always hold true hence, it is considered as naivs Bayes. Since the features are assumed

to be independent given class, from the rule of probability, joint model can be expressed as:

$$p(C_i|\mathbf{X}) = \frac{p(x_1|C_i) \times p(x_2|C_i) \times ... \times p(x_n|C_i) \times p(C_i)}{p(\mathbf{X})}$$

The term $p(\mathbf{X})$ at denominator can be considered as constant term as it does not depend upon C_i , thus can be ignored. .

Now, the naive Bayes classifier with *maximum a posterior* or MAP decision rule can be written as:

$$\text{Classify}_{naive\,Bayes}(x_1, x_2, ..., x_n) = \arg\max_c p(C_i = c) \prod_{i=1}^{n} p(X_i = x_i|C_i = c)$$

In above formulation, class conditional probabilities, i.e., $p(X_i = x_i|C_i = c)$ need to be estimated from the training data. For this, naive Bayes treats discrete and numeric feature values differently. For each discrete feature, $p(X = x|C = c)$ is calculated by first making the frequency table for each feature value for given class label and dividing it by frequency of instances with same class label and then using the naive bayes equation. And in case of numeric features, they are assumed to follow some continuous probability distribution over the range of that feature's value. Commonly, naive Bayes assumes numeric features follow the Gaussian distribution. Although, many real world data follows this distribution, this assumption might not hold true in some domains. In such case, if we know that features follows some other distribution, we can use that distribution for probability estimation. However, if we are not sure about the actual distribution of features, a method called Kernel estimation can be used to approximate the probability. (John and Langley, 1995) have shown that in numbers of natural fields, naive Bayes classifier with kernel estimation has shown better classification result than naive Bayes that assumes Gaussian distribution.

Naive Bayes is simple yet successful classifier. It does not require large amount of data for training. Although, its assumption of feature independence is generally a poor assumption, naive Bayes often competes well with more sophisticated classifiers (Rish, 2001).

c) Artificial neural network

Human brain consists of 100 billions neurons. Each neuron is connected with other thousands of neurons and communicate via electrical signals. When neuron receives multitude of signal, it is combined in some way, and, if combination is strong enough (above certain threshold), it fires

output signal to other interconnected neurons.

Similar to how biological neural network works, artificial neural network (ANN), a network of many simple units tries to imitate it to process information through mathematical models that are capable of pattern recognition due to its adaptive behaviors. ANN is interconnected group of artificial neurons. An example of artificial neuron is shown in 2.3 A neural network

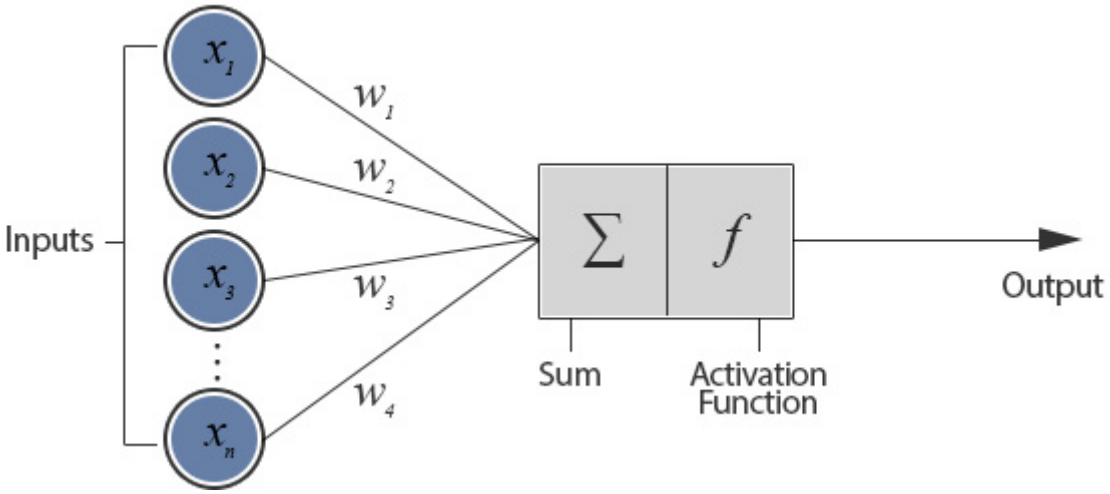


Figure 2.3: Artificial neuron[3]

representation of perceptron consists of multiple inputs nodes and a output node. Each input is associated with corresponding weight. These inputs and their corresponding weight are multiplied and then summed together. This summation of weighted input is called an activation where activation is then passed through function called as activation function or transfer function (which is generally is a simple threshold activation function). The transfer function then compares the activation value with threshold to produce the output. If activation is higher than threshold, higher valued output generally "1" is outputted at output node, and if activation is lower than threshold, "0" is outputted. In order to get the correct output from neuron, suitable weight to each input nodes should be found. And there is a method to find the suitable weights to each input, and it is called *perceptron learning rule*. In supervised classification, all the inputs have desired output. Given initialized vector of weight, perceptron learning rule try to iteratively adjust the weight vector to get the desired output for each input from the network. This process of adjusting the weight is called learning or training of network.

Perceptrons are single layer artificial neural network. They consists of only two layers: input and output layers. Perceptrons are able to classify the data only if they are linearly separable

[3]http://www.theprojectspot.com/images/post-assets/an.jpg

(Huang, 2009). In order to explain complex decision boundary, multiple perceptrons can be used as building block to form larger and more complex network architecture known as *multi-layer perceptrons* (MLP) network. MLP network consists of input layer which receives the training instances, one or more hidden layer which receives the output from previous layer and weight them and pass through activation function (usually non-linear), and output layer which takes output form last hidden layer and produces output. Hidden layers are called so because they have no input or output to external environment. Figure 2.4 shows the architecture MLP network consisting of two hidden layers .

Training of MLP network for classification consists of two problems. First, finding the suitable network architecture like how many hidden layers or how many nodes in each layer, and second being how to adjust the weight parameter. Generally, in order to find the suitable network architecture for given problem, different network architecture are trained and the one which gives lowest classification error is chosen as final network model. After network structure is fixed, the problem lies in finding the suitable weights. As a solution to solve the problem of learning the weights of MLP network, an algorithm known as *back-propagation algorithm* is developed by (Rumelhart et al., 1986). The algorithm looks for minimum value of error function with respect to weights in network using a gradient descendant optimization technique. The weight that minimizes the error function is considered as optimal weights for the network.

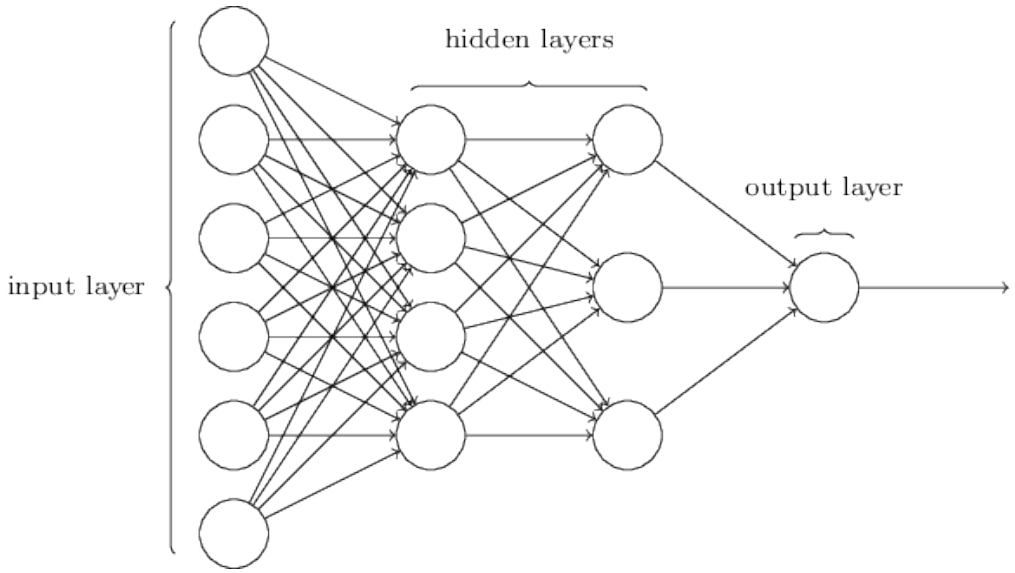


Figure 2.4: multilayer perceptron network[4]

[4]http://neuralnetworksanddeeplearning.com/images/tikz11.png

d) Ensemble learning

Ensemble learning is assembly of multiple classifiers or models that solve the same original task. In ensemble classification scheme, classification decision made by different classifiers are integrated using some method to classify new instances. Ensemble learning methods have become one of the active area of research in supervised learning (Dietterich, 2000). Of many ways of learning ensemble of classifiers, prominent among them are *bagging*, *boosting*, and *stacking* (Witten and Frank, 2011).

To introduce, in bagging classifiers are trained with training sets consisting of instances that are randomly sampled with replacement from original training set. Such newly generated training set is called *bootstrap replicate* of original training set (Dietterich, 2000). Usually, the size of original training set and bootstrap replicate are equal. Since sampling is done with replacement, it might happen that some of the instances might get repeated multiple times while some of them might not get selected at all. Now, to classify the new instances, each classifier returns its prediction and final prediction is done based on majority vote count, i.e, composite classifier assigns the test instance the class label that was predicted most often.

Boosting is another general ensemble method for increasing the predictive performance of any learning algorithm (Rokach, 2009). Boosting works by iteratively running weak classifiers and then combine them into single strong composite classifier. Assigning weight to each training instances forms the basis in boosting. Initially, all the instances in training set are equally weighted. Later, weight changes in each round of learning based on whether the training instances are correctly classified or misclassified. Boosting algorithms call this each round of learning "weak" learning. If a instance is correctly classified, its weight is decreased, but if it is misclassified, its weight is increased. As a result of this, following weak learner is forced to focus on the higher weighted training instances or in other words, instances that are difficult to classify correctly. Boosting share the similarity with bagging in a way that both use similar classifier for combining. However, unlike in bagging where classifiers are independent of each other, in boosting classifiers complement each other. Moreover, rather than giving a equal weight to each model as in bagging, in boosting contribution of each model is weighted by its confidence (Witten and Frank, 2011).

Stacking also called as stacked generalization is another ensemble techniques that aims to achieve higher generalization accuracy (Wolpert, 1992). It is not as frequently used as bagging and

boosting though. In stacking, generally the models of different types are used. The idea behind stacking is that, first various models are learned using the original training set. And new data set is created using prediction values returned by these models for each instances, and true values of each instances. This new data set is now used as input to another model. In original paper of stacking, original data and models learned using it at first step are called as *level 0 space* and *level 0 generalizers*, respectively, and data and model learned in second step are termed as *level 1 data* and *level 1 generalizers*. Now, we will discuss about one of the ensemble learning algorithm called random forest.

Random forest (RF) proposed by (Breiman, 2001) is an ensemble learning method involving the ensemble of another learning technique called decision tree. In RF, decision trees are trained pre-defined number of times using the training set comprising of the bootstrap samples of original training set. Hence, random forest utilizes bagging technique explained earlier. In addition, m number of variables among M variables in original training set are randomly selected at each node and best split on these m is used to split the node. Finally, in order to classify the test instance, majority rule is used, i.e., the test instances is assigned the class label which is predicted most often by the classification trees.

It is shown that the error rate of RF depends upon correlation between any two trees and strength of each individual tree in random forest (Breiman, 2001). Increased correlation increases the error rate while increasing the strength decreases the error rate. And these two measure depend upon the value of m: the number of variables randomly sampled at each split. Increasing m increases both and vice versa. Hence, while training the random forest classifier, "m" is the parameter that needs to be tuned in order to have better classification performance of RF model. Along with value of m, another parameter that needs to be tuned is number of trees to be grown in forest. It needs to be tuned because optimal value of it guarantees stable and robust result.

2.2 Dimensionality reduction

Many real word dataset are of high dimension. High dimensional dataset means dataset with large number of features or attributes. One typical example of high dimensional dataset in bioinformatics is microarray gene expression dataset, where expression level of thousands of genes are measured in a single experiment. With high dimensionality of data, many problems

may arise in downstream analysis of it including 1) needing larger memory space to store it, 2) visualization of data is not possible, 3) excessive time in analyzing the data, and 4) the curse of dimensionality. “Curse of dimensionality” refers to the phenomena that as the dimensionality of data increases, the number of data points needed increases exponentially (Bellman, 1961).

To address the problem of the higher dimensionality of data, various dimensionality reduction techniques have been proposed. Dimensionality reduction is a technique to reduce the dimension of data by removing the noisy features. There are two kinds of dimensionality reduction techniques: feature selection and feature extraction. Feature selection deals with the selection of subset of features from original features (Dash and Liu, 1997) while in feature extraction, new feature space is created of lower dimension usually as combination of original features, in contrast to feature selection where no such feature transformation takes place (Saeys et al., 2007; Tang et al., 2014). So, in feature extraction, physical meaning of original feature is lost but this remains intact in feature selection. In regards to this, feature selection is superior in term of better readability and interpretability (Tang et al., 2014).

This thesis work focuses on feature selection process and more detail is provided in below section, but for now following paragraph discusses about one of the feature extraction method called as principal component analysis. Principal component analysis (PCA) is one of the most simplest and well-known dimensionality reduction algorithm. PCA performs linear transformation on a data set by rotating the coordinates to obtain new coordinate system called principal components (PCs) with the goal such that the first PC captures the highest variance (hence, information) from data, second PC captures second highest variance and so on. The first step involved in PCA is to create the covariance matrix. Then after the eigenvectors and eigenvalues are deduced from covariance matrix. The eigenvectors are sorted in descending order in relation to their eigenvalues which forms the new coordinate or PCs. As said earlier, as the PCA is linear transformation of data, all the resulting PCs are linear combination original variables.

2.3 Feature selection

Feature selection is one of the most important and widely used data preprocessing techniques in data mining task like classification, clustering, regression and association rules (Sutha and Tamilselvi, 2015). Feature selection, also known as attribute selection or variable selection, is

a process of removing irrelevant, redundant or noisy data to get a subset of relevant features (Kumar and Minz, 2014; Liu and Yu, 2005). Irrelevant features are those features which carries no useful information in describing the data, and redundant features are those features which add no information than provided by already present features (Zeng et al., 2015). Feature selection has numbers of advantages. It enhances result comprehensibility, lowers computational cost and increases the data mining performances like classification accuracy(Kumar and Minz, 2014; Liu and Yu, 2005; Tang et al., 2014). Feature selection process can be done in both supervised and unsupervised learning. However, we will discuss feature selection process in the context of supervised learning (classification) problems as this thesis work is focused on classification task. In classification task, feature selection tries to select subset of highly discriminatory features such that they have a capability in distinguishing the samples that belong to various classes (Tang et al., 2014).

2.3.1 Basic steps of feature selection

A general feature selection procedure involves following four basic steps. A unified view of feature selection process is shown in figure 2.5. Brief description to each step is provided below.

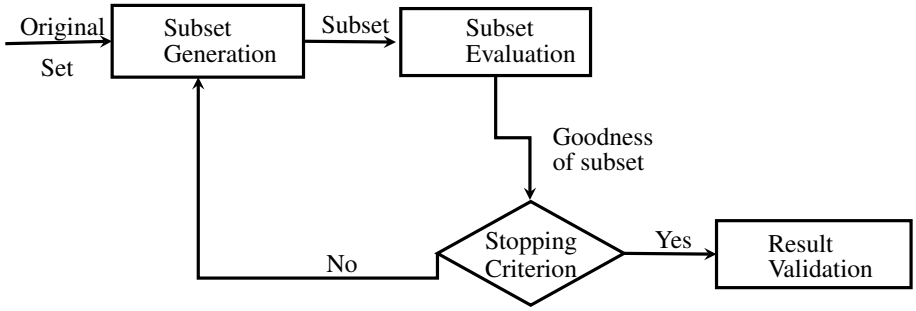


Figure 2.5: Key steps in feature selection adopted from (Liu and Yu, 2005)

a) Subset generation

This is the first step in feature selection where a feature subset is generated for evaluation. For a data set with N numbers of features, the number of possible subsets to be evaluated is 2^N. Hence, even for a moderate N, this number of search space becomes prohibitive for exhaustive search. Therefore, different search strategies are devised: complete, sequential, and random search.

Complete search method guarantees to find the optimal subset according to the criterion used.

While an exhaustive search is complete (i.e., no optimal subset is missed), a search does not have to be exhaustive in order to guarantee completeness (Liu and Yu, 2005). For this, different heuristic functions are implemented to decrease the search space without compromising the chance of finding the optimal subset. Hence, even though order of search space is $\mathcal{O}(2^N)$, fewer subsets are evaluated (Liu and Yu, 2005; Dash and Liu, 1997; Kumar and Minz, 2014).

Sequential search gives completeness risking losing the optimal subset(Liu and Yu, 2005). There are different variants of sequential search like sequential forward search, sequential backward elimination and bi directional search. All these search methods either add or remove one or more feature at time even though initially selected subset might be different as there might be zero or all features in selected list. Sequential searches are fast to produce result as the order of the search space is $\mathcal{O}$ (N^2) or less (Liu and Yu, 2005).

Random search begins with randomly selected subset and moves forward in either of two ways. First being sequential search discussed above, but injecting randomness in it, and second way is by generating next subset in complete random manner. The concept of randomness is to avoid local optima in search space and optimality of subsets depends on the resources available (Liu and Yu, 2005; Kumar and Minz, 2014).

b) Subset evaluation

In this step newly generated subset from subset generation step is evaluated by certain evaluation criteria. The optimal subset is always relative to certain criterion such that a subset selected to be as optimal subset by one criterion may not be optimal subset when evaluated by another criterion (Dash and Liu, 1997; Liu and Yu, 2005). After evaluation, if newly generated subset is better than previous subset, it replaces previous subset as best subset .

c) Stopping criteria

This step guides when feature selection process should stop. There are various feature selection stopping criteria. The general feature selection stopping criteria are predefined maximum number of iterations, minimum error rate is achieved or predefined minimum numbers of features in final subset or addition/deletion of features do not produce significantly better subset (Liu and Yu, 2005; Kumar and Minz, 2014).

d) Validation

In this final step, the chosen subset is validated either by prior knowledge or validation set. When we have domain knowledge on relevant features, we expect these features to be in chosen subset. With real-world data, we often lack this knowledge, in such case performance measure like classification error can be used to compare the result of chosen subset to that of using all features.

2.3.2 Categories of feature selection algorithms

Based on how feature selection process incorporates classification model, feature selection technique can be divided into three categories: filter, wrapper, and embedded methods. A brief introduction to each of these method is presented below.

a) Filter method

Filter method selects a subset of feature based on general characteristic of data without using any learning algorithm to measure the goodness of feature subsets. A general schematic diagram of filter method based feature selection is shown in figure 2.6. From figure, we can see feature selection is performed prior to introduction of learning algorithm.

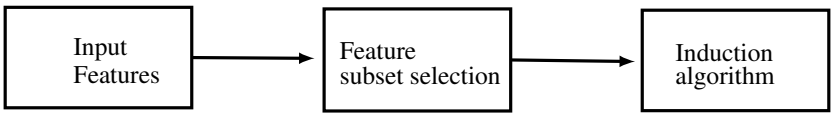


Figure 2.6: a schematic diagram of filter methods of feature selection from (Kohavi and John, 1997)

Filter methods of selecting features can be further divided into two groups based on whether they evaluate features individually or as group forming feature subset. These two groups are univariate methods and multivariate methods.

- Univariate methods: Univariate feature selection methods deal with each individual feature to determine the strength of relationship of feature with output variable ignoring the interaction among features. Univariate methods of feature selection are feature ranking method. Feature ranking assign weight to each features independent of other features and rank them based on their relevance to the target concept (Yu and Liu, 2003). As an output, all input features are given relevance score and user can then set certain threshold to select

Generalized filter algorithm from (Liu and Yu, 2005)

Input:$D(F_0, F_!, ...F_n - 1)$ // a training data set with N feature
S_0 // a subset from which to start the search
δ // a stopping criterion
output: S_{best} // optimal subset

```
1: begin
2:     initialize: S_best = S_0;
3:     γ_best = eval(S_0, D, M); //evaluate S_0 by an independent measure M
4:     do begin
5:         S=generate (D); //generate a subset for evaluation
6:         γ=eval(S,D,M); // evaluate the current subset S by M
7:         if (γ is better than γ_best)
8:             γ_best=γ;
9:             S_best=S;
10:    end until (δ is reached)
11:    return S_best;
12: end;
```

required number of features or select pre-defined number of top-ranked features. Since in univariate methods features are ranked individually based on their relevancy to output class, feature redundancy is completely neglected (Saeys et al., 2007; Yu and Liu, 2003). However, empirical evidence from the feature selection literature shows that along with irrelevant features, redundant features should also be removed as redundant features affect both the accuracy and computational time of machine learning algorithms (Hall, 1999; Yu and Liu, 2003; Kohavi and John, 1997). The advantages of univariate methods are such that these methods take less time to run, simple to understand and provide better understanding of data. Examples of univariate feature selection methods are t-test, Chi-squared test, and methods based on mutual information.

- Multivariate methods: These methods consider the interaction among features during feature selection process in contrast to univariate methods. While in univariate methods, features are individually ranked, in multivariate methods, algorithm searches for possible feature subsets and evaluate them using certain criteria. Hence, unlike in univariate methods, feature redundancy and interdependencies are considered in multivariate methods, and as a result redundant features may be excluded from optimal subset (Guyon and Elisseeff, 2003). The disadvantages of multivariate methods are that they are slow and less scalable than univariate methods (Saeys et al., 2007) Correlation based feature selection (CFS) (Hall, 1999) and Markov blanket filter (Koller and Sahami, 1996) are examples of

multivariate feature selection.

Filter methods of feature selection are computationally simple and fast and easily scalable to high dimensional data (Saeys et al., 2007; Sutha and Tamilselvi, 2015). Also, as feature sets selected by filter methods are independent of learning algorithms, different classifiers can be trained with same feature set. However, the main disadvantage of filter methods is that they completely ignore the effects of selected subset on the performance of the induction algorithm (Kohavi and John, 1997).

b) Wrapper methods

Wrapper methods based feature selection use performance of pre-determined learning algorithm as a criteria to evaluate the feature subset. In wrapper methods, learning algorithm is taken as "black box" (Kohavi and John, 1997). A schematic diagram of wrapper method based feature selection is shown in 2.7. In these methods, different possible subsets are generated through a specific search algorithm and the specific learning algorithm is trained using each subset. The performance of classifier trained with each subset is evaluated and the subset which gives best performance is chosen as optimal subset.

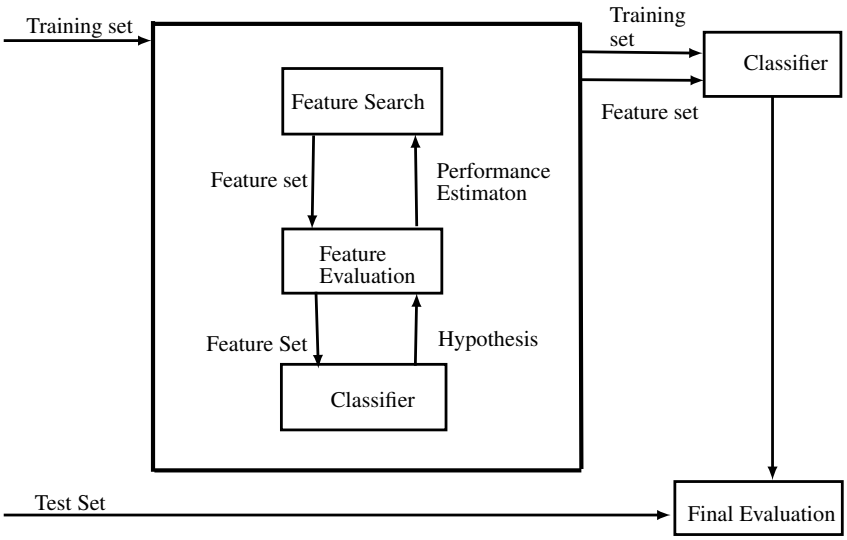


Figure 2.7: a schematic diagram of wrapper methods of feature selection from (Tang et al., 2014)

Advantages of wrapper methods are such that the feature interdependencies, and interaction between feature subset and model selection are well considered (Saeys et al., 2007). Wrapper methods generally yields better predictive performance estimates than filter methods (Kohavi and John, 1997).

Next to the advantages, there are also disadvantages of wrapper approach. Wrapper methods are computationally expensive than filter methods which is due to the need to train classifier to each generated feature subset (Saeys et al., 2007; Langley et al., 1994). Also, this approach has higher chance of overfitting than filter methods(Saeys et al., 2007). Wrapper methods lack the generality as optimal subset of features selected is optimal to specific learning algorithm only (Saeys et al., 2007; Tang et al., 2014).

Generalized wrapper algorithm from (Liu and Yu, 2005)

Input:$D(F_0, F_!, ...F_n - 1)$ // a training data set with N feature
S_0 // a subset from which to start the search
δ // a stopping criterion
output: S_{best} // optimal subset

```
1: begin
2:     initialize: S_best = S_0;
3:     γ_best = eval(S_0, D, A); //evaluate S_0 by mining algorithm A
4:     do begin
5:         S=generate (D); //generate a subset for evaluation
6:         γ=eval(S,D,A); // evaluate the current subset S by A
7:         if (γ is better than γ_best)
8:             γ_best=γ;
9:             S_best=S;
10:    end until (δ is reached);
11:    return S_best;
12: end;
```

c) Embedded method

In Embedded feature selection methods, feature selection is taken as a part of classifier construction. In contrast to filter method where no learning algorithms are involved, and in wrapper method where learning algorithms are used to measure the goodness of feature subsets, in embedded methods, learning part and feature selection part can not be separated (Lal et al., 2006). A schematic diagram of embedded feature selection is shown in 2.8.

Embedded method uses the statistical criteria to select the various subsets with given cardinality just like in filter methods, and uses the accuracy of classifier to select final optimum subset just like in wrapper methods (Tang et al., 2014). Thus, embedded methods possess the advantages of (1) filter methods- they are far less computationally cheaper than wrapper methods, and (2) wrapper methods - they incorporate the feature and model interaction (Saeys et al., 2007), and have comparable classification accuracy (Tang et al., 2014). In term of drawback, embedded

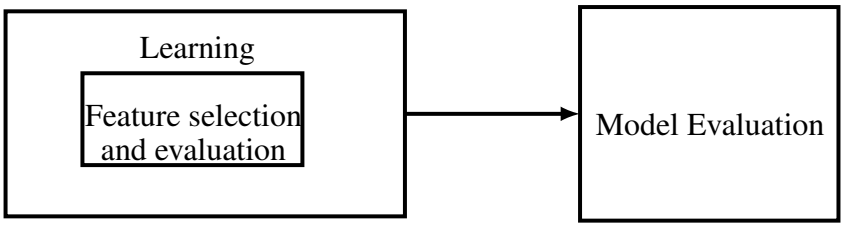


Figure 2.8: a schematic diagram of embedded methods of feature selection from (Hilario and Kalousis, 2008)

methods share similarity with wrapper methods. Similar to wrapper methods, features selected by embedded methods are also learning algorithm dependent (Saeys et al., 2007).

Generalized embedded algorithm from (Liu and Yu, 2005)

Input:$D(F_0, F_1, ...F_n - 1)$ // a training data set with N feature
S_0 // a subset from which to start the search
output: S_{best} // optimal subset

begin
  initialize: S_{best} = S_0;
  c_0 = card(S_0); //calculate the cardinality of S_0
  γ_{best}= eval(S_0,D,M); // evaluate S_0 by an independent measure M
  θ_{best} = eval(S_0,D,M); // evaluate S_0 by a mining algorithm A
  for c = c_0 +1 to N **begin**
    for i = 0 to N - c **begin**
      S = $S_{best} \cup F_j$; // generate a subset with cardinality c for evaluation
      γ = eval (S,D,M); // evaluate the current subset S by M
      if (γ is better than γ_{best})
        γ_{best} = γ;
        S'_{best} = S;
    end;
      θ = eval(S'_{best}, D, A); // evaluate S'_{best} by A
      if (θ is better than θ_{best});
        S_{best}= S'_{best};
        θ_{best}= θ;
      else;
        break and **return** S_{best};
  end;
  return S_{best};
end

2.4 Feature selection algorithms implemented in this project

This section explains the feature selection algorithms used in this thesis project. Four different feature selection methods, namely Minimum Redundancy Maximum Relevance (mRMR), variable

selection with Random Forest and the area under the curve (AUCRF), least absolute shrinkage and selection operator (LASSO), and the wrapper methods were used. General introduction to each of these methods is presented below.

a) AUCRF

Area under the ROC curve of random forest (AUCRF)(Calle et al., 2011) is a feature selection algorithm based on optimizing the area-under the ROC curve (AUC) of random forest. This algorithm implements the backward elimination process based on initial ranking of features. The algorithm first construct the random forest model using all the features. The variables are then ranked and specified fraction of least important variables are eliminated. The random forest model is again constructed with remaining variables and AUC value is computed. This process is repeated until the remaining number of variables is less than or equal to specified value. Finally, feature subset which gives highest AUC value to the random forest is considered as optimal feature subset.

b) mRMR

minimum Redundancy Maximum Relevance (mRMR) is a feature selection algorithm proposed by (Ding and Peng, 2005; Peng et al., 2005). This algorithm tries to overcome the aspect of feature redundancy by supplementing usual maximum relevance criteria with minimum redundancy criteria during feature selection process. mRMR requires features to be maximally dissimilar to the one which is already identified as relevant features before choosing that feature. As a result, features selected by mRMR are expected to be more representative of target class leading to better generalization accuracy (Liu et al., 2008).

mRMR feature selection algorithm used in this thesis is adpoted from (De Jay et al., 2013). Implementation follow such that let y be the output variable and X =$\{x_1$,x_2,....,$x_n\}$ be set of n input features and S be the selected subset of features, mRMR method ranks X by maximizing the mutual information (MI) with y (maximum relevance) and minimizing the average MI with all previously selected features (minimum redundancy). Feature with highest MI with y is selected

first and S is initialized with this feature. If x_i is the feature with highest MI, it can be written as

$$x_i = argmax_{x_i \in X} I(x_i, y) \tag{2.7}$$

where I denotes the mutual information information between feature x_i and output variable y. In similar fashion, next feature added to S is the one which has highest relevance with y and lowest redundancy with previously selected feature/s, thus maximizing the score q at step j

$$q_j = I(x_i, y) - \frac{1}{|S|} \sum_{x_k \in S} I(x_j, x_k) \tag{2.8}$$

where $I(x_j,x_k)$ is mutual information between features x_j and x_k.

c) LASSO

Least absolute shrinkage and selection operator (LASSO) (Tibshirani, 1996) is an algorithm that employs regression analysis method to perform variable selection. LASSO sets the coefficient of some of the predictor variables equal to zero by using $L1$ penalty. Given the outcome variable y_i, for cases i=1,2,...n and features x_{ij} where j=1,2,...p, LASSO minimize

$$\sum_{i=1}^{n} (y_i - \sum_j \beta_j x_{ij})^2 + \alpha \sum_{j=1}^{p} |\beta_j| \qquad \text{subject to } \sum |\beta_j| \leq s \tag{2.9}$$

where $s \geq 0$ is parameter which controls the strength of penalty and it needs to be tuned . As the value of this parameter increases, fewer variables are selected, i.e more coefficient are reduced to zero and there is more shrinkage of non zero coefficient.

d) Wrapper methods

A wrapper method of feature selection involves generating the different subsets of features and evaluating the goodness of each feature subset using performance measure of classification algorithm. Finally, the feature subset which has highest classification performance is chosen as optimal feature subset for given classifier. In this project, Best-first search (BFS) method is used to search for optimal subset of feature and four different classification algorithms, namely näive Bayes, K-nearest neighbor, random forest, and support vector machine with radial basis function

kernel are used to evaluate the feature subsets. Best-first search is chosen to conduct the search because it has shown to be perform superiorly than greedy hill-climbing method (Kohavi and John, 1997).

2.5 Performance measures

Once a classification model is learned, evaluating its performance is another step in classification problem. This evaluation is done based on how accurately model classifies the samples in test set. And the number of samples that are either correctly or incorrectly classified can be tabulated in a matrix called as confusion matrix as shown in table 2.1.

Table 2.1: Confusion matrix for binary classification

	Predicted: YES	Predicted: NO
Actual: YES	TP	FN
Actual: NO	FP	TN

A confusion matrix is a contingency table in which rows correspond to true class and columns to predicted class. Now, explaining the every element in confusion matrix.

- TP: Number of samples which are positive and classified as positive (true positive).
- FN: Number of samples which are positive but classified as negative (false negative).
- FP: Number of samples which are negative but classified as positive (false positive).
- TN: Number of samples which are negative and classified as negative (true negative).

Although, confusion matrix itself provides the general information on how good model is, various other performance measure metrics can be calculated based on it. Among other, accuracy is one of the simplest and widely used performance measure metric, and is defined as ratio of correctly classified samples to that of total number of samples in test set. Since the diagonal elements of confusion matrix are the number of correctly classified samples, overall accuracy of classifier is given by:

$$\text{Accuracy} = \frac{\text{Number of correctly classified samples}}{\text{Total number of samples}} = \frac{TP + TN}{TP + TN + FP + FN}$$

Sometime, instead of classification accuracy, classification error or error rate is also used to quantify the performance of classifier which can be calculated as:

$$\text{Error rate} = \frac{\text{Number of incorrectly classified samples}}{\text{Total number of samples}} = \frac{FP + FN}{TP + TN + FP + FN}$$

Equivalently, error rate can be expressed also as:

$$\text{Error rate = 1 - Accuracy}$$

Even though accuracy or error rate alone can be used to assess the performance of model, this performance metric alone does not provide information on how accurate is model if dataset is highly imbalanced. Here, imbalanced dataset means majority of samples belongs to either one of the class. Suppose there is a dataset whose 90% of samples belongs to positive class and remaining 10% to negative class. A classifier which classifies all the samples to positive class would achieve the accuracy of 90 % which makes classifier apparently a good classifier even if it has failed to correctly classify even a single stance of negative class. Thus, in order to have robust and better idea about classification performance, there are two class-dependent performance measures that are typically used in medical diagnosis and they are sensitivity and specificity.

Sensitivity is a measure of proportion of positive-class samples correctly classified as such. It is also known as recall or True positive rate (TPR) depending up on field of application. In the context of this thesis, sensitivity is defined as proportion of metastatic samples that are actually classified as metastatic. Sensitivity can be formulated as

$$Sensitivity = \frac{TP}{TP + FN}$$

Similarly, *Specificity* (also know as True negative rate) is measure of the proportion of negative-class samples correctly classified as such. In this thesis, it is proportion of non-metastatic region that are correctly classified as such. Specificity is given by

$$Specificity = \frac{TN}{TN + FP}$$

Another popular performance measure used in binary classification problem is based on receiver operating characteristic (ROC) curve (Fawcett, 2006). ROC curve is two-dimensional graph

created by plotting true positive rate (TPR) against the false positive rate (FPR). TPR is already defined above where as FPR can be defined as:

$$FPR = \frac{FP}{TN + FP} = 1 - Specificity$$

The ROC curve space is shown in figure 2.9. The dashed diagonal line from bottom left to top right denotes random classifier performance. To consider any classification useful, operating point has to be above this line. Similarly, if classifier falls below this line, it is consider worst than random guessing. ROC curve can used to quantify the performance of binary classifier by calculating the scalar performance metric known as area under the curve (AUC). Exact value of AUC can be obtained by integrating the ROC curve. The AUC value lies between 0 and 1, and higher the value of AUC, the better the performance of classifier is. A perfect classifier has the AUC value of 1, which is at the point (0,1) at ROC space. In addition, since the ROC curve is plot of sensitivity vs 1- specificity for various values of classifier parameter, it can be used to find the desired balance between sensitivity and specificity for given classifier by selecting appropriate parameter value.

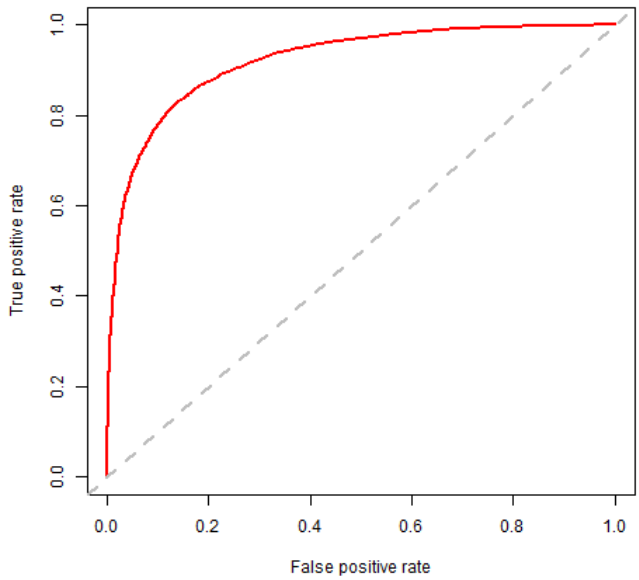


Figure 2.9: ROC space

2.6 Methods to generate training and testing set

In classification problem, with some exception, all learning algorithms have one or more free parameter(s) to learn. Now, the questions arise how to select the optimal parameter(s) of a model, and once the model is selected, how to estimate its performance for given problem ? Straightforward answer to these questions would be to train the model on entire available dataset and estimate the error rate and choose the model which has lowest error rate. However, this approach has basic problem in a form that the error rate estimate would be overly optimistic and also this does not give information on how good is a model in predicting the new instances, i.e, instances that are not used in learning the parameters of model. Therefore, in practice dataset is randomly split into two disjoint subsets: *training set* and *test set*. This method of generating two subsets of data is called *Holdout method*. In holdout method, generally data is split in 70:30 ratio for training and testing. Thereafter, training set is used to train the model, and test set is used to estimate the error rate of trained model. Even though holdout method is much better approach than using entire dataset for training classifier and estimating the error rate of model, this method has drawback. Since it is a single run of splitting of dataset, this could lead to misleading estimate of error rate if there is "unfortunate split" of data. In addition, this method does not fully utilizes the available data points for training the classifier as considerable chunk of data needs to be set aside for testing.

To overcome the limitations of holdout method, re-sampling based approach has been proposed. In re-sampling based approach, training and testing set are generated multiple times with random samples where samples are drawn without replacement and this forms the basis for what is called a *cross-validation* (Witten and Frank, 2011). The most common approaches in cross validation are k-fold cross validation and leave-one-out cross validation "LOOCV". In k-fold cross validation, available dataset is partitioned into k folds where each fold is approximately of equal size. Out of k folds, k-1 folds are used to train the model and remaining fold is used to estimate the error rate. This whole process is repeated k times and final error rate is the averaged error rate over k folds. Commonly chosen number of fold is 10, which is then called as 10-fold cross validation. Similarly, leave-one-out cross validation is special case of k-fold cross validation where k equals the number of samples in dataset. For a dataset with N samples, N experiments of training and testing of classifier is carried out. In each experiment, N-1 instances are used to train the classifier and remaining instance to test it. Although, cross

validation techniques are computationally expensive, they give better approximation of error rate of classifier as each observation is used both for training and testing the classifier.

2.7 Pattern recognition for lymph node metastasis detection

Computer assisted diagnosis (CAD) in histopathology is young and emerging field (Gurcan et al., 2009). Such emergence in research activities in this field is mainly due to development in high throughput whole slide digital imaging techniques capable of producing high-resolution digital images (Pantanowitz et al., 2011; Sertel, 2010). Similarly, various pattern recognition techniques have been applied to detect the lymph node metastasis in breast cancer from whole slide images. The result page of ISBI camelyon challenge shows participants using different techniques like Random forest, support vector machine, logistic regression, neural network, deep learning for detecting lymph node metastasis in breast cancer [5].

The use of pattern recognition in detecting and predicting the metastasis in lymph node section of breast cancer patients is not confined to based on whole slide images only, but also in combination with various other clinical, pathologic or biological features. (Takada et al., 2012) used decision tree based method known as alternating decision tree (ADTree) to predict axillary lymph node (AxLN) metastasis in primary breast cancer where they achieved the ROC AUC value of 0.772 using clinicopathological features. Likewise, (Wu et al., 2014) used various tumor biological parameters that included age, tumor size, grade, estrogen receptor, progesterone receptor, lymphovascular invasion, and HER2 to train a support vector machine for predicting axillary lymph nodes (ALN) metastases where they achieved the accuracy of 74.7 % in correctly predicting ALN metastases. Other techniques used in detecting metastases in breast cancer includes (Lancashire et al., 2008) applying multi-layer perceptron in gene microarray dataset to identify and validate gene signatures corresponding with estrogen receptor and axillary lymph node status in breast cancer achieving 100% accuracy in later case.

[5] https://camelyon16.grand-challenge.org/results/

Similarly, (Marchevsky et al., 1999) evaluated 19 prognostic features through neural networks (NN) with genetic algorithms to predict the status of axillary lymph node. In the same study, logistic regression model was also used for the same task, however, using only 6 features. The accuracy of the neural network fitted with 240 cases and logistic regression model with 240 were 89.0 and 66% respectively. Another study utilizing genetic algorithm and multilayer perceptron (MLP) for predicting axillary Lymph Node (ALN) status was done by (Karakış et al., 2013). In the study, genetic algorithm was used for selecting best features and optimizing weights of backpropagation algorithm in MLP. With 9 features based on clinical, radiological, and pathological examinations, they predicted axillary lymph node status with the accuracy of 98.0% Here, the general overview of different pattern recognition technique used in detection of lymph node detection in breast cancer is shown. As seen, different study have used different dataset utilizing different features.

2.8 Histipathology - a review

Histopathology is study of the signs of the disease using the microscopic examination of a biopsy or surgical specimen that is processed and fixed onto glass slides (Gurcan et al., 2009). During the process, samples taken from patients are sliced into minimally thin slices, which are then placed on glass slides and examined under the microscope. During microscopic examination, structure of tissues under study is studied at various level and decision is made based on how distinctly these structure share similarities between normal vs. abnormal tissue.

Before being analyzed under the microscope, thin segment of biopsy samples are stained with various pigments. The biopsy samples are stained because most of the living tissues look colorless, sometime even transparent under microscope making it difficult to distinguish structural details of the tissues(Kårsnäs, 2014). Now, with staining, it gives color to colorless tissues samples providing contrast between cellular and extra-cellular components (Sertel, 2010). Commonly, tissues are dyed with two types of stains. First, principal stains are used to highlight cellular components and at same time; counter stains with contrasting color to principal stain are used to increase contrast. Among various staining methods, the most widely used stain in histopathology, either for diagnostic or research purpose is Hematoxylin-Eosin (H&E) (McCann et al., 2015; Kårsnäs, 2014; Gurcan et al., 2009). Hematoxylin stains cell nuclei making them appear blue, while Eosin stains cytoplasm and connective tissue making them appear pink

(Gurcan et al., 2009).

Along with technological advancement, use of innovative digital imaging methods in pathology is on the rise(Farahani et al., 2015). Among different digital imaging techniques, whole slide imaging (WSI), a relatively novel method, is a process of producing the digital slides of traditional glass slides using high resolution scanners (Pantanowitz et al., 2011; Farahani et al., 2015). This digitization of glass slides now allows the examination of pathological specimen in computer with the help of image viewer. Recently, WSI has gathered worldwide interest among pathologists for education, research, and diagnostic purposes (Farahani et al., 2015).

As mentioned in introductory section of thesis, a computer assisted approaches can be used to analyze the histopathological slides. And computer assisted diagnosis of histopatological images consists of mainly three different computational steps (Demir and Yener, 2005): 1) Processing of images to determine the focal areas, 2) Feature extraction that could quantify the properties of focal areas, and 3) Diagnosis. Step one involves determining the area of interest or focal areas which usually preceded by noise removal to improve image quality in order to enhance its success. In second step, various quantitative features that characterizes area of interest is extracted and in the final step, an automated system is developed using different machine learning techniques, including measures for estimating the accuracy of developed system, in order to perform actual diagnosis.

3. Aims and Objectives

In the light of need for automated assessment of histopathological images, the International Symposium of Biomedical Imaging (ISBI) held the Camelyon Grand Challenge 2016 (Camelyon 2016) [1]. The goal of the challenge was to evaluate the new and existing algorithms for automatically detecting the metastases in hematoxylin and eosin (H&E) stained whole-slide images of lymph node sections in breast cancer. A research group from Faculty of Medicine and Life Sciences at University of Tampere had also took part in the grand challenge [2]. This thesis work is a part of research work done by that group.

While the computational steps involved in pre-processing of whole-slide images and feature extraction are not within the scope of this thesis work as it was already done by aforementioned research group, this research focuses on developing a framework of machine learning techniques to detect metastatic region of lymph node section in breast cancer. The schematic diagram of proposed framework is shown in figure 3.1.

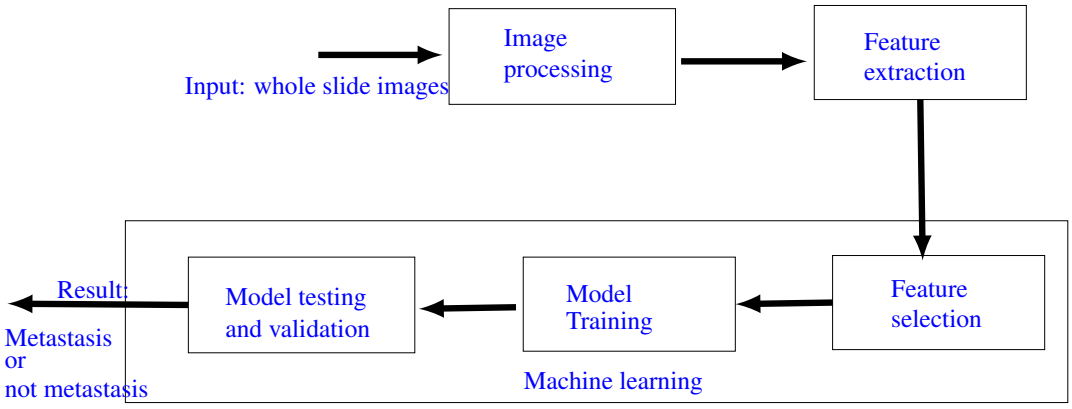


Figure 3.1: Framework for classification of lymph node metastasis

The main goal of this research project is to find the most relevant features among features extracted from whole slide images for classification between non-metastatic and metastatic lymph node region. Following research objectives are set to achieve the goal.

1. To identify the most relevant features using different feature selection techniques

[1] https://camelyon16.grand-challenge.org/
[2] http://www.uta.fi/bmt/institute/research/nykter/index.html

2. To develop the classification system using selected features for classification between normal and metastatic region

3. To evaluate and verify the developed system

4. Material and methods

4.1 Material

4.1.1 Data

Data used in this thesis were extracted from 47 whole slide images (30 normal + 17 tumor slides). Each sample in dataset represents 200x200 pixel (full resolution) blocks that were randomly sampled from the tissue area. And from each sample block 214 features that describe the image morphology, texture and spatial distribution of nuclei were extracted. For further information about pre-processing of whole slide images and feature extraction part, (Valkonen et al., 2017) can be reviewed.

4.2 Methods

4.2.1 Feature selection

a) AUCRF

This is a feature selection algorithm using random forest based on optimizing AUC value of random forest. This algorithm performs the backward elimination of variables, where it first runs random forest using all features and removes the specified number of least important features based on important score provided by random forest model. Next, it again runs the random forest using remaining variables and calculates the AUC value. This process is repeated until the desired number of remaining features is reached. This algorithm is publicly available as R package called "AUCRF" (Urrea and Calle, 2012). R is a software environment for statistical computing and graphics (R Core Team, 2016). In this thesis work, algorithm was ran until there was only one feature left after elimination process removing single feature at a time.

b) LASSO

Proposed by (Tibshirani, 1996), LASSO is shrinkage and feature selection algorithm. According to (Kim and Kim, 2004), LASSO achieve better prediction accuracy by shrinkage as the ridge regression, but at the same time, also gives a sparse solution, which means that some of the coefficients are exactly 0, thus performing variable selection. In this project, package "glmnet" (Friedman et al., 2009) available in R was used for implementing the LASSO algorithm. The tuning parameter lambda was selected using 10-fold cross validation. Feature selection was performed by fitting the binomial classification scheme.

c) mRMR

Minimum redundancy maximum relevance (mRMR) is a feature selection algorithm which aims at maximizing the feature relevance while trying to minimize the feature redundancy. Feature relevance and redundancy is measured using the mutual information which is computed using approximation based on correlation. During the project work, mRMR feature selection was performed using R package "mRMRe"(De Jay et al., 2013), in which mutual information is estimated as

$$I(x, y) = -\frac{1}{2}\ln(1 - \rho(x, y)^2) \tag{4.1}$$

where I and ρ represent the mutual information and correlation coefficient between variables x and y respectively.

d) Wrapper method

For the wrapper method, Best-first search algorithm was used to generate the different feature subsets and four different classifiers were used to evaluate the subsets to find the optimal subset for each classifier. The training data were internally divided into training and testing set. Then, classification algorithms were trained with training set with different feature subset and testing set was used to estimate the performance. Classification accuracy was as a performance measure metric. The feature subset which yielded the highest classification accuracy was chosen as optimal subset. The classification algorithms were again ran on whole training set with optimal subset and their performance was evaluated on independent test set that was not used for searching optimal feature subset. Since, BFS algorithm allows to back-track to more promising previous

subset if the node being analyzed at present appears to be less promising, maximum number of such allowed back tracking was set to be 5.

4.2.2 Classifiers

a) K-nearest neighbors (KNN)

K- nearest neighbors is an algorithm based on applying nearest neighbor rule that assigns test instance the class label which is most common among its k nearest neighbors in training set. Nearness among samples is measured by specific distance metric. In this thesis, in order to find best value of k, different odd values of k ranging from k=1 to k =$\sqrt{n}$ were tried, where n is size of training set. And the value of k which gave lowest classification error was chosen as the optimal value.

b) naive Bayes

It is a simple probabilistic classifier with strong assumption of each feature being independent. In this thesis, instead of assuming normal distribution of training set for estimating the probability density function, kernel density estimation was used.

c) Logistic regression

Logistic regression, also called as logit model, is a regression technique used to fit the model when the response variable is dichotomous (binary), i.e, output variable Y_i= 1 or 0 (an event happens or not). It has no requirement over the distribution of independent variables. If p is the probability of response variable Y_i to be 1 given multiple predictors x_1,x_2,x_3,x_4,...x_k, logistic response function is defined as:

$$\frac{p}{1-p} = e^{\beta_0+\beta_1 x_1+\beta_2 x_2+\beta_3 x_3+...+\beta_k x_k} \tag{4.2}$$

where the term $\frac{p}{1-p}$ is knows as odd ratio of event. Now, taking natural logarithm on both sides of above equation

$$log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + ... + \beta_k x_k \tag{4.3}$$

Equation 4.3 is the equation used in modeling logistic regression. If the left hand side of it, i.e, if the log of odd ration is positive, the probability of success (event happening) is always more than 50%.

In this project, logistic regression model was fitted by specifying parameter "family"= "binomial" in glm() function available in R statistical software environment.

d) Random forest

Random forest (RF) is an ensemble learning method comprising of decision trees. There are two parameters that needs to be tuned while building RF model: number of trees to be grown and the number of variables randomly sampled at each split (m). Generally, the default value to try for "m" for classification task is considered to be square root of number of variables in training set. However, in this thesis work, different values of m (m=1 to m=square root of number of variables in training set) were tried. Similarly, RF was trained with different tree sizes of (25,51,101,201,301,...1001). While tuning for the optimal parameters, Out-Of-Bag (OOB) error was used as criteria. In RF, OOB error gives the unbiased test set error. During the construction of each tree in RF, about one third of samples are left out of bootstrap samples. These left out samples are called out of bag samples. OOB error is the classification error of out of bag samples.

e) Suport vector machine

In this thesis, SVM with two types of kernels were studied: linear and radial basis function(RBF). There are two free parameters associated with these kernels: regularization parameter (C) and gamma (γ). gamma ($\gamma > 0$) controls the width of RBF kernel. The accuracy of SVM classification highly depends upon the values of these two parameters. With higher value of C and/or gamma (γ), SVM overfits to the training data and classifier is not generalizable. So, suitable values for these parameter needs to be determined. The recommended practical way to find the optimal value of these parameters is to try out a "grid search" on exponentially growing

sequences of C and γ (Hsu et al., 2003). So in order to find the optimal value of C for linear kernel, grid search was used over the space of C = $(2^{-5}, 2^{-3}, ...2^{15})$ and for RBF kernel, two dimensional grid search was used where the space of search for C was as mentioned above and gamma was searched over γ=$(2^{-15}, 2^{-13}, ...2^{3})$.

f) Back propagation neural network (BPNN)

This is the implementation of multi-layer neural network trained with back-propagation algorithm. Activation function used in hidden layer was sigmoid. Two hidden layers with each consisting of 10 neurons were used as network architecture. Furthermore, for two main parameters, namely learning rate(η) and momentum (β) which need to be tuned while training the BPNN (Amato et al., 2013; Bengio, 2012), the value of momentum parameter was selected 1, i.e., no momentum was used as it produced the best result, and for η, different values from $10^{-}6$ to 1 were tried. In addition, neural network was trained with batch size of 32 samples.

5. Results and Discussion

This chapter presents the results obtained from different classification schemes used for automated detection and classification of metastases using the features extracted from whole-slide images stained with hematoxylin and eosin (H&E) of lymph node section in breast cancer. Also, the results of data visualization are presented in here.

5.1 Features selection and classification results

Here, we present the feature selection and classification results of different feature selection techniques implemented in this project work. Feature set obtained from different feature selection techniques were used to train the different classifiers and four different performance measure metrics: accuracy, area under the ROC curve (ROC AUC), sensitivity, and specificity were used to assess the performance of classifiers.

Now, the results section begins with presenting the classification result of mRMR feature selection algorithm. With this algorithm 40 features were selected, and seven different classifiers were trained using those features. Classifiers that were trained are naive bayes, random forest, logistic regression, multi-layer perceptron, and support vector machine with linear (SVM-Linear) and radial basis function kernel (SVM-RBF). Performance of these classifiers were evaluated by two methods: hold-out and 10-fold cross validation. The classification result is visualized in the form of box plot where it is possible to see how the different performance measure scores are distributed in each fold, and on the top of it, performance measures of hold out technique and average of 10 fold cross validation measures are also shown for each classifier.

Foremost, boxplot in 5.1 presents the accuracy of various classifiers. From the figure, it can be seen that random forest classifier has highest classification accuracy followed support vector machine with radial basis function kernel, albeit the difference between them is very minimal. And lowest prediction accuracy is of naive bayes classifier.

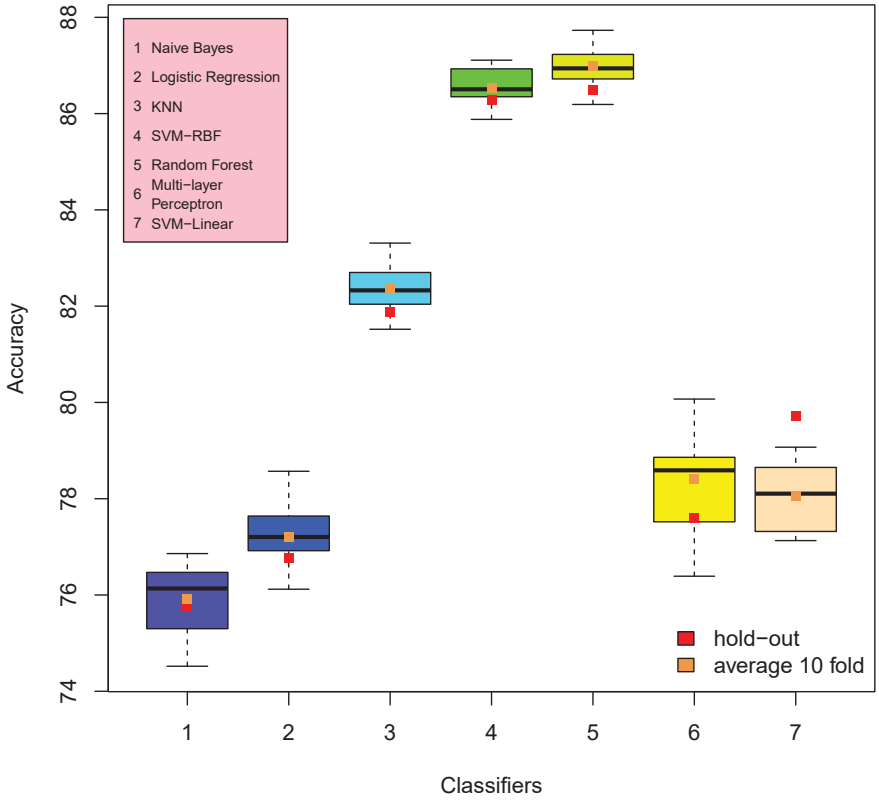


Figure 5.1: Accuracy of various classifiers induced with mRMR feature selection

Likewise, figures in 5.2 and 5.3 show the ROC AUC values and sensitivity (i.e., fraction of metastatic samples correctly classified as such) of various classifiers, respectively.

In automated detection of metastasis in lymph node, it is highly desirable to have higher sensitivity. Higher sensitivity ensures that true metastasis does not get missed out from detection, which is important in breast cancer prognosis. However, higher sensitivity should not come at the cost of much reduced specificity. If there is low specificity, there might be chance that non metastatic patient get diagnosed as such. Figure 5.4 shows the specificity (i.e fraction of non metastatic samples correctly classified as such) of various classifiers. From the box plots of sensitivity and specificity, it can be seen that there is good balance between sensitivity and specificity for all most all the classifiers, even though some classifiers have higher sensitivity and

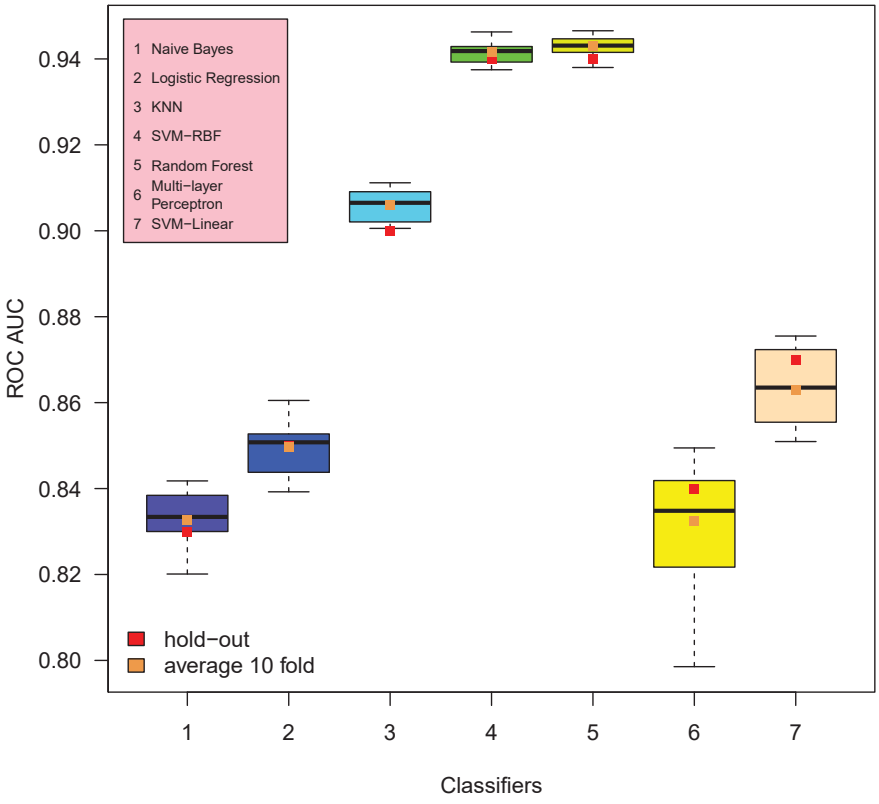


Figure 5.2: ROC AUC values of various classifiers induced with mRMR feature selection

some have higher specificity. Only the multi-layer perceptron, a aritifical neural network model showed the higher discrepancy between sensitivity and specificity.

Furthermore, another information that the above four box plots provide is how the performance measures metric scores are distributed over each fold in 10-fold cross validation. From the box plots, it can be seen that performance measures metric scores in each fold do not vary significantly for all most all of the classifiers as the length (height) of each box is comparatively shorter. Among different classifiers, neural network has shown to have higher variability in measurement across 10-fold cross validation, whereas random forest model has least variability. The reason why random forest classifier showing less variability in performance measure can be traced back to fact that it is an ensemble learning method involving many decision trees. In addition, in each box plot, a hold-out accuracy is also shown on it. Given the minimal variation across 10-fold cross validation accuracy, and hold-out accuracy lying within range of average 10-fold cross validation, it is safe to assume that hold-out performance measures alone can be taken as unbiased performance estimate for this data set.

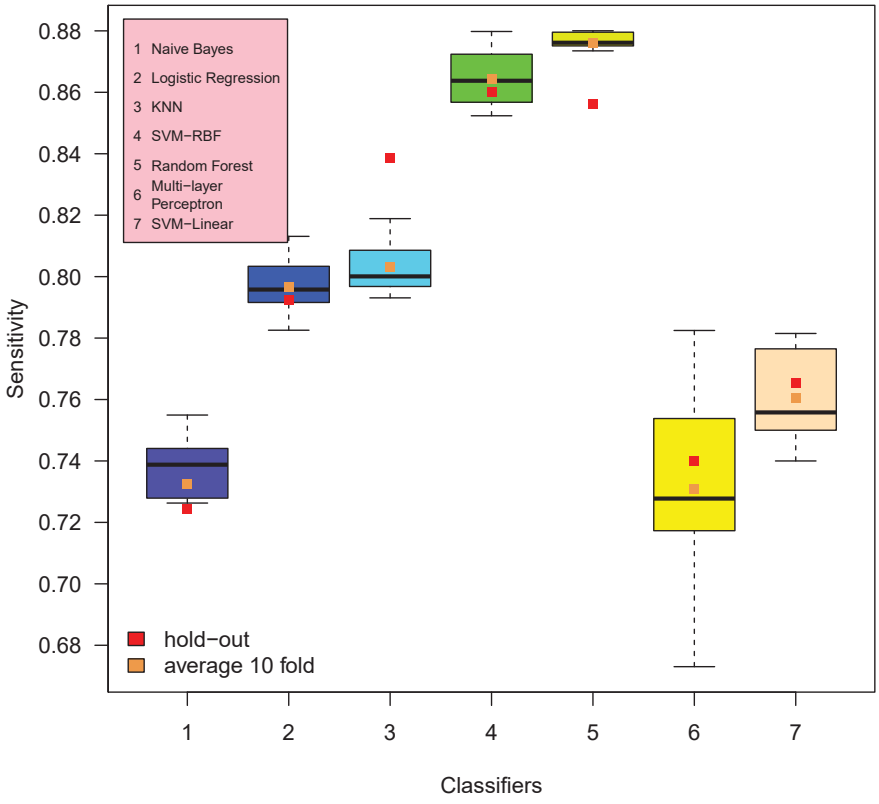


Figure 5.3: Sensitivity of various classifiers induced with mRMR feature selection

Similarly, feature selection results of three other feature selection methods wrapper, AUCRF - a feature selection algorithm based on optimizing the area under the curve of random forest, and LASSO feature selection based on logistic regression feature selection techniques is presented now. In wrapper method, BFS algorithm was employed to search for the feature subsets and four different classifiers were used to evaluate the subsets. The classifiers that were used in wrapper methods are naive Bayes (NB), K-nearest neighbor (KNN), random forest (RF), and support vector machine with radial basis function kernel (SVM-RBF). Classification result of all these methods is tabulated in 5.1 When comparing the classification performance of various

Table 5.1: Classification result of wrapper, AUCRF, and LASSO feature selection

Feature selection	**Accuracy (%)**	**ROC AUC**	**Sensitivity**	**Specificity**
BFS + NB	74.74	0.82	0.78	0.70
BFS + KNN	83.68	0.91	0.86	0.80
BFS + RF	88.31	0.95	0.87	0.88
BFS + SVM-RBF	87.69	0.95	0.87	0.87
AUCRF	89.60	0.96	0.89	0.89
LASSO	83.02	0.90	0.83	0.83

classifiers in two different cases: one induced after mRMR feature selection and another with

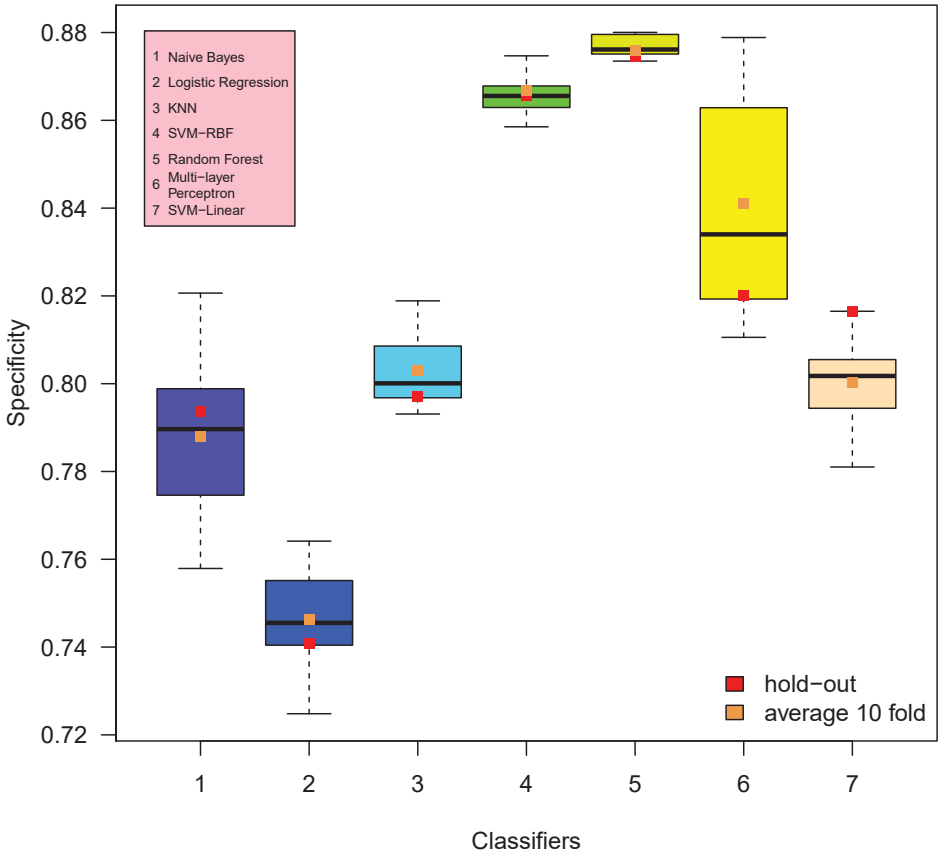


Figure 5.4: Specificity of various classifiers induced with mRMR feature selection method

wrapper, AUCRF and LASSO feature selection, it can be seen that, except for naive bayes, there is increase in overall classification performance for all classifiers with later three feature selection. Wrapper, AUCRF and LASSO showing improved results over mRMR do not come as surprise because these methods directly incorporate performances of learning algorithms to determine the goodness of feature subset in order to find the optimal feature subset. In other words, in case of Wrapper, AUCRF and LASSO, feature selection is optimized for specific learning algorithm which is involved into feature selection process. The exact number showing the classification results of selected classifiers after mRMR feature selection is shown in table 5.2. The highest difference in classification performance is noticed with logistic regression classifier. And in the case of naive bayes classifier, there is slight decrease in overall accuracy and ROC AUC value with wrapper feature selection than with mRMR. Sensitivity is higher with wrapper feature selection and specificity is higher with mRMR feature selection. This change in sensitivity and specificity of naive bayes with respect to feature selection technique can be used depending upon

the application when more sensitivity or specificity is needed.

Table 5.2: Classification result of mRMR feature selection

Classifier	**Accuracy (%)**	**ROC AUC**	**Sensitivity**	**Specificity**
Naive Bayes	75.76	0.83	0.72	0.79
K- nearest neighbor	81.87	0.90	0.84	0.79
Random forest	86.5	0.94	0.85	0.87
SVM-RBF	86.28	0.94	0.86	0.86
Logistic regression	76.77	0.85	0.79	0.74

Now, the number of features in optimal subset for wrapper, AUCRF and LASSO feature selection method is shown in table 5.3

Table 5.3: Number of features in optimal set for different feature selection methods

Feature selection	**Number of features in optimal subset**
BFS + NB	5
BFS + KNN	6
BFS + RF	13
BFS + SVM-RBF	19
AUCRF	39
LASSO	144

From the above table, in addition with number of feature selected by mRMR algorithm which was 40, it can be seen that different feature selection technique has different number of feature in their optimal set. This diversity is due to the fact that different feature selection techniques use different criteria to evaluate the feature. Furthermore, in wrapper feature selection, where same technique was used to generate the feature subset, but used different classification algorithm to evaluate them, it can be seen that different classifier has generated feature space of different dimension, i.e., selected different number of features. This feature space of varying dimension generated by different classifier relates to the fact that different classifier uses different approach to treat feature space for classification (like naive bayes uses probabilistic rule, whereas KNN uses distance measures), and each classifier can has its own feature space where its performance is optimum. Hence, it can be said that, for classification, it is not necessary to have same feature space to obtain optimal classification performance when using different classification algorithms.

5.1.1 Validating feature selection on independent data set

After performing the classification with different feature selection techniques, here the validation of feature selection method is carried out on independent data set. Feature selection is validated by "before - after experiment", where the performance of the classifiers are evaluated twice: once when whole feature feature set is used to train the classifiers and next when only selected feature set is used to train them. Finally, these two results are compared. First of all, table 5.4 summarizes the classification performance of two classifiers — random forest, and K-nearest neighbor — before performing feature selection. Detail classification results of various classifiers using same dataset can be found in (Valkonen et al., 2017)

Table 5.4:
Classification performance estimation on
validation data set before feature selection (in %)

Classifier	**Accuracy**	**Sensitivity**	**Specificity**
Random forest	92.63	93.33	91.83
K- nearest neighbor	85.68	89.4	82.27

Now, the table in 5.5 shows the classification performance after wrapper and AUCRF feature selection.

Table 5.5:
Classification performance estimation
of different feature selection methods on validation data set (in %)

Feature selection	**Accuracy**	**Sensitivity**	**Specificity**
BFS+RF	91.77	93.03	90.4
AUCRF	93.92	94.64	93.12
BFS+KNN	87.90	91.0	84.44

Comparing the performance of classifiers before and after feature selection from tables 5.4 and 5.5, it can be seen that classification performance enhanced with all feature selection methods except in the case of wrapper feature selection with random forest, where accuracy decreased slightly. Even though accuracy decreased slightly in that case, it was not degraded. Now, looking into how much feature space has been reduced due to these feature selection methods, table 5.6 shows the reduction in feature space after different feature selection techniques.

Table 5.6: Reduction in feature space after feature selection

Feature selection	**% reduction in feature space**
BFS+RF	93.93
AUCRF	81.77
BFS+KNN	97.19

From above results, we can now say that feature selection has bought twofold advantage. First, it has helped to improve the classification results, and second, even if it didn't improve the classification performance in one case, there is massive reduction in dimensionality of data without compromising performance of classifiers. This reduction in dimensionality of data greatly helps in reducing the classifier training time, and also helps effective training of it. Apart from these discussed benefits that feature selection bought in terms of classifiers performance and reduction in their computational time, another advantage that feature selection brings can also be looked in terms of cost associated with feature extraction from whole slide images in future problems. As feature selection now provides the prior knowledge about which features and/or what minimum number of features are beneficial for predictions, having these information avoid extracting more features unnecessarily. And this might be of great importance in application where there are thousands of whole slide images to process, and quicker prediction decisions are needed to make.

5.1.2 Analyzing the selected features

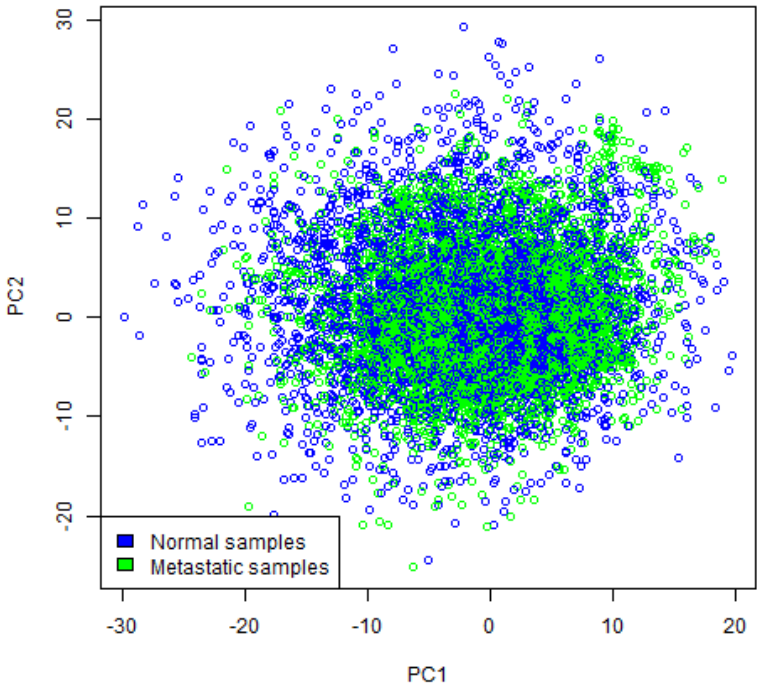
Total of 168 different features got selected by different feature selection methods. Highest number of features was selected by LASSO and was 144, and lowest number was by best first search with naive bayes and was 5. While analyzing the most frequently selected features among 168 feature, it was found that non of feature was selected by all feature selection techniques. Only one feature was selected by 6 feature selection methods, 3 features were selected by 5 feature selection methods, 5 features got selected exactly by 4 methods, 16 features by exactly 3, 32 feature by exactly two and remaining by either one of the method. Overall, 57 features were selected by more than one methods and 111 by one of the method. However, it should be noted that among 111 features, LASSO alone selected 105 features and remaining 6 features were selected by other methods. From above discussion it can be seen that there are not many features that are selected by majority of feature selection methods. This diversity in final feature

subset selected by different feature selection methods reflects the redundancy among the features in original feature set. As different feature selection technique implements different criteria to evaluate the feature subset, one specific feature might has got selected among the different features that shares redundancy with it. In addition, such analysis of overlapping features helps not only to find the most salient features among the selected features, but further interpretation of most common features can also be carried out. Since feature selection keeps the physical meaning of selected feature intact, further analyzing salients features can be interesting and may even come up with revealing information about the problem being studied.

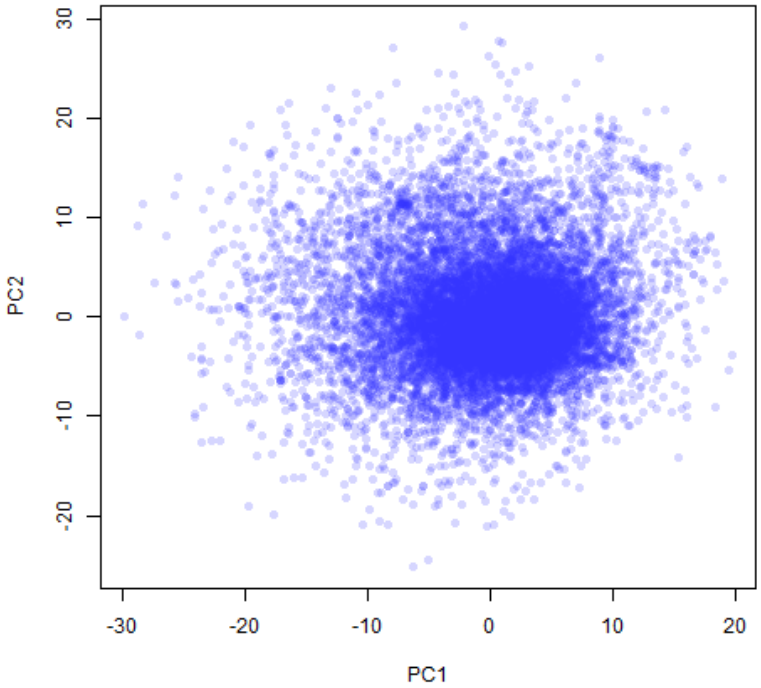
5.2 Data visualization

This section explains about the data visualization, which is done in two cases: before and after feature selection. Principal component visualization was used to visualize the data. First and second principal components (PCs) can be used to visualize the data in two dimension. As principle components try to preserve as much data information as possible, visualizing the samples using the PCs gives the general idea about the separability of samples into their respective group. Here, in the context of this thesis, it should be noted that data visualization does not belong to feature selection steps or automated detection of metastasis pipeline. Therefore, even though various feature selection methods were implemented, feature selected by only one feature selection method was used for visualizing the data.

Scatter plot in figure 5.5 shows the cancer samples along first two PCs. Upper panel of figure reveals that first two PCs clearly fails to separate the samples into different classes. Even after coloring the samples according their class labels, there is no distinct clusters formed between the normal and metastatic samples. Furthermore, lower panel of same figure, where samples are plotted as transparent points, shows that many normal and metastatic samples do overlap each other.



(a) PCA of samples with samples colored according to their class labels



(b) Each sample plotted as transparent points. Darker the color, the more overlapping samples

Figure 5.5: PCA visualization of samples before feature selection

Likewise, scree plot in figure 5.6 shows the information (variance) carried by PCs. From the figure, it can be seen that first two PCs explains about 30 % of variance. From further analysis, it is found that it takes around 150 PCs to explain 99 % percent of variance. As 99% variance of data can be explained by 150 PCs, it means that intrinsic dimensionality of data is lower than the dimension of the original dataset, i.e., actual number of variables present in dataset.

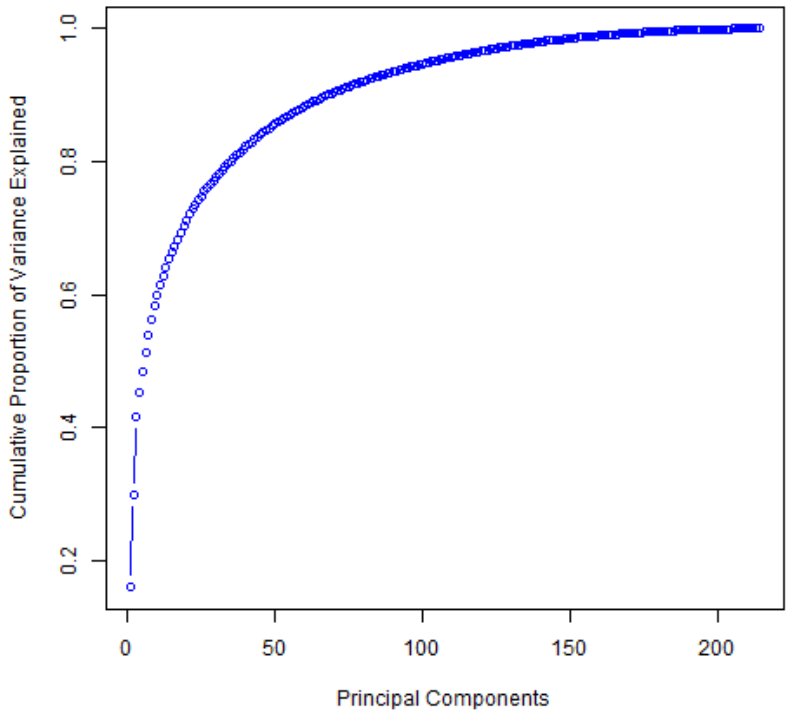
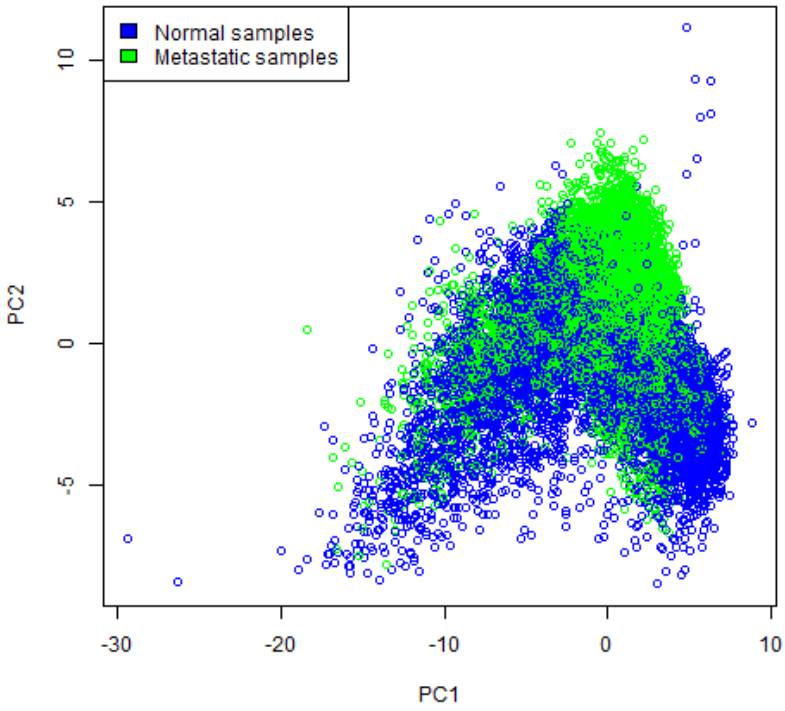
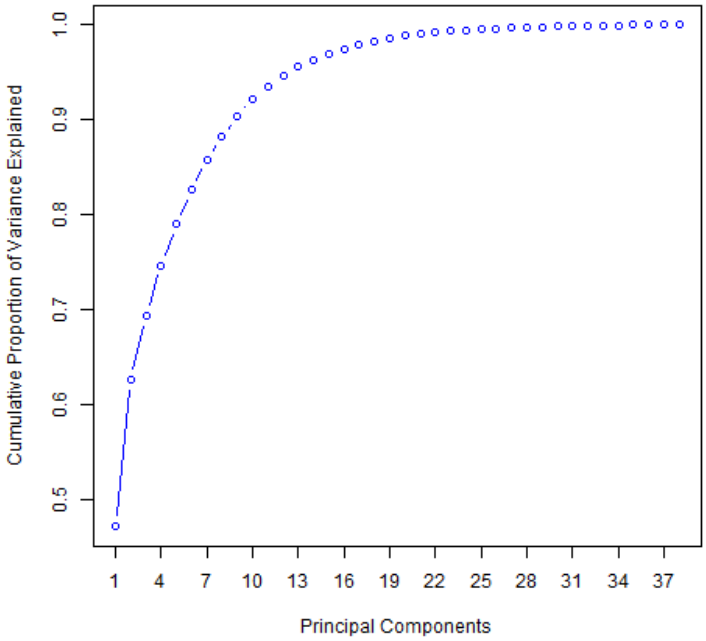


Figure 5.6: Scree plot showing the cumulative proportion of variance explained by PCs before feature selection

Further, in subsequent section, we would see the PCA visualization after feature selection. Feature selected by AUCRF was used for PCA visualization. It should be noted that PCA is unsupervised dimensionality technique, but in this case it can not be said so as features that are used in PCA are selected by supervised feature selection method. Figure 5.7 shows PCA visualization after feature selection along two PCs. Upper panel of figure shows the samples across two PCs. From the figure, it can be seen that in contrast to PCA visualization before feature selection, PCA visualization after feature selection shows some cluster as some metastatic samples grouped together, even though samples are still not linearly separable, and are also overlapped. Hence, it is not surprising that this PCA visualization shows grouped samples as PCA space was created with more discriminating features.



(a) PCA of samples after feature selection with samples colored according to their class labels



(b) Scree plot showing the cumulative proportion of variance explained by PCs after feature selection

Figure 5.7: PCA visualization of samples after feature selection

Now, the lower panel of the figure 5.7 shows the variance explained by PCs after feature selection. From figure, it can be seen that first two PCs explain more than 60% of variance. As first two PCs explains 60% of variance, it means data is more compressed, i.e, higher information content with reduced dimension. Furthermore, from variance plot we can see that there is no significant increase in variance explained or variance explained by PCs becomes stable after 20th PC. This suggests that even though dimensionality of data has been reduced to 39 features after feature selection, this dimension can still further be decreased without losing much information.

6. Conclusion

This thesis work focused on development of framework for automated detection of lymph node metastases in breast cancer with the features extracted from digitized tissue images. Automated detection system is based on application of machine learning algorithms, where machine learning algorithms are trained to build the classifiers with set of features. Many features might be extracted from images, but not all of them may contribute in classifying between metastatic and non metastatic samples.

Various feature selection methods were implemented for identifying the relevant features from original set of features extracted from whole slide images. mRMR is multivariate feature selection method that incorporates the feature interaction unlike univariate methods. Different classifiers were build using the features selected by mRMR. Among the different classifier, random forest and support vector machine with radial basis function kernel performed better than others. Along with mRMR, wrapper methods, AUCRF and LASSO feature selection were also implemented. And their performance were better than mRMR. In addition, classification performance of different classifiers were assessed in two different cases: when they were trained with full feature set, and with reduced feature set. This assessment of classification performance of different classifiers showed the advantages of feature selection process in classification problems. The advantages were that the performance of classifiers were enhanced with reduced feature set, and even if performance didn't enhance in certain case, computational time of algorithm was reduced without degrading the performance. Furthermore, to identify the most relevant features among the feature selected by different feature selection techniques, analysis of features selected by different methods was carried out. This analysis showed that there are few features that were selected by majority of feature selection methods revealing the redundancy present among features in original feature set.

Furthermore, PCA was also performed showing the variance explained by PCs in data set. Variance explained by PCs before feature selection showed that maximum variance in data can be accumulated using number of PCs lower than total number of variables present in data set, which further vindicated the feature selection process. Further, two dimensional PCA visualization of

data was also done. While the data visualization before feature selection revealed the nature of data such that there was no distinct cluster formed between metastatic and non-metastatic samples, visualization after feature selection showed some cluster as more informative features were used for visualization,demonstrating the advantage brought by feature selection process.

To conclude, the results obtained from the implementation and evaluation our proposed framework demonstrate that the automated computer assisted detection of metastases from digitized tissue images is feasible and efficient. These results look promising in the light of need for automated detection of metastasis; either in cancer research or in clinical application, where a successful implementation would significantly lower both the subjectivity of traditional manual interpretation of histopathological images and the workload of pathologist arising due to this manual handling of histopathological slides.

Bibliography

Al-Kofahi, Y., Lassoued, W., Grama, K., Nath, S. K., Zhu, J., Oueslati, R., Feldman, M., Lee, W. M., and Roysam, B. (2011). Cell-based quantification of molecular biomarkers in histopathology specimens. *Histopathology*, 59(1):40–54.

Alitalo, K. and Carmeliet, P. (2002). Molecular mechanisms of lymphangiogenesis in health and disease. *Cancer cell*, 1(3):219–227.

Amato, F., López, A., Peña-Méndez, E. M., Vaňhara, P., Hampl, A., and Havel, J. (2013). Artificial neural networks in medical diagnosis.

Bellman, R. E. (1961). *Adaptive control processes - A guided tour*. Princeton University Press, Princeton, New Jersey, U.S.A.

Bengio, Y. (2012). Practical recommendations for gradient-based training of deep architectures. In *Neural networks: Tricks of the trade*, pages 437–478. Springer.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Calle, M. L., Urrea, V., Boulesteix, A.-L., and Malats, N. (2011). Auc-rf: A new strategy for genomic profiling with random forest. *Human heredity*, 72(2):121–132.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.

Cunningham, P. and Delany, S. J. (2007). k-nearest neighbour classifiers. *Multiple Classifier Systems*, 34:1–17.

Dash, M. and Liu, H. (1997). Feature selection for classification. *Intelligent data analysis*, 1(3):131–156.

De Jay, N., Papillon-Cavanagh, S., Olsen, C., El-Hachem, N., Bontempi, G., and Haibe-Kains, B. (2013). mrmre: an r package for parallelized mrmr ensemble feature selection. *Bioinformatics*, 29(18):2365–2368.

Demir, C. and Yener, B. (2005). Automated cancer diagnosis based on histopathological images: a systematic survey. *Rensselaer Polytechnic Institute, Tech. Rep*.

Dietterich, T. G. (2000). Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer.

Ding, C. and Peng, H. (2005). Minimum redundancy feature selection from microarray gene expression data. *Journal of bioinformatics and computational biology*, 3(02):185–205.

Farahani, N., Parwani, A., and Pantanowitz, L. (2015). Whole slide imaging in pathology: advantages, limitations, and emerging perspectives. *Pathol. Lab Med. Int*, 7:23–33.

Fawcett, T. (2006). An introduction to roc analysis. *Pattern recognition letters*, 27(8):861–874.

Fortmann-Roe, S. (2012). Understanding the bias-variance tradeoff. `http://scott.fortmann-roe.com/docs/BiasVariance.html`. [Online; accessed 19-March-2017].

Friedman, J., Hastie, T., and Tibshirani, R. (2009). Lasso and elastic-net regularized generalized linear models. in: R package.

Gurcan, M. N., Boucheron, L. E., Can, A., Madabhushi, A., Rajpoot, N. M., and Yener, B. (2009). Histopathological image analysis: A review. *IEEE reviews in biomedical engineering*, 2:147–171.

Guyon, I. and Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182.

Hall, M. A. (1999). *Correlation-based feature selection for machine learning*. PhD thesis, The University of Waikato.

Hanahan, D. and Weinberg, R. A. (2000). The hallmarks of cancer. *cell*, 100(1):57–70.

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition (Springer Series in Statistics)*. Springer.

Hilario, M. and Kalousis, A. (2008). Approaches to dimensionality reduction in proteomic biomarker studies. *Briefings in bioinformatics*, 9(2):102–118.

Hsu, C.-W., Chang, C.-C., Lin, C.-J., et al. (2003). A practical guide to support vector classification.

Huang, Y. (2009). Advances in artificial neural networks–methodological development and application. *Algorithms*, 2(3):973–1007.

Jirina, M. and Jr (2011). *Classifiers Based on Inverted Distances*. INTECH Open Access Publisher.

John, G. H. and Langley, P. (1995). Estimating continuous distributions in bayesian classifiers. In *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, pages 338–345. Morgan Kaufmann Publishers Inc.

Karakış, R., Tez, M., Kılıç, Y., Kuru, Y., and Güler, İ. (2013). A genetic algorithm model based on artificial neural network for prediction of the axillary lymph node status in breastcancer. *Engineering Applications of Artificial Intelligence*, 26(3):945–950.

Kårsnäs, A. (2014). Image analysis methods and tools for digital histopathology applications relevant to breast cancer diagnosis.

Kim, Y. and Kim, J. (2004). Gradient lasso for feature selection. In *Proceedings of the twenty-first international conference on Machine learning*, page 60. ACM.

Kohavi, R. and John, G. H. (1997). Wrappers for feature subset selection. *Artificial intelligence*, 97(1):273–324.

Koller, D. and Sahami, M. (1996). Toward optimal feature selection.

Kumar, V. and Minz, S. (2014). Feature selection: A literature review. *SmartCR*, 4(3):211–229.

Lal, T. N., Chapelle, O., Weston, J., and Elisseeff, A. (2006). Embedded methods. In *Feature extraction*, pages 137–165. Springer.

Lancashire, L. J., Rees, R. C., and Ball, G. R. (2008). Identification of gene transcript signatures predictive for estrogen receptor and lymph node status using a stepwise forward selection artificial neural network modelling approach. *Artificial intelligence in medicine*, 43(2):99–111.

Langley, P. et al. (1994). Selection of relevant features in machine learning. In *Proceedings of the AAAI Fall symposium on relevance*, volume 184, pages 245–271.

Liu, H. and Yu, L. (2005). Toward integrating feature selection algorithms for classification and clustering. *IEEE Transactions on knowledge and data engineering*, 17(4):491–502.

Liu, J., Ranka, S., and Kahveci, T. (2008). Classification and feature selection algorithms for multi-class cgh data. *Bioinformatics*, 24(13):i86–i95.

Marchevsky, A. M., Shah, S., and Patel, S. (1999). Reasoning with uncertainty in pathology: artificial neural networks and logistic regression as tools for prediction of lymph node status in breast cancer patients. *Modern pathology: an official journal of the United States and Canadian Academy of Pathology, Inc*, 12(5):505–513.

McCann, M. T., Ozolek, J. A., Castro, C. A., Parvin, B., and Kovacevic, J. (2015). Automated histology analysis: Opportunities for signal processing. *IEEE Signal Processing Magazine*, 32(1):78–87.

Metter, G. E., Nathwani, B. N., Burke, J. S., Winberg, C. D., Mann, R. B., Barcos, M., Kjeldsberg, C. R., Whitcomb, C. C., Dixon, D., and Miller, T. P. (1985). Morphological subclassification of follicular lymphoma: variability of diagnoses among hematopathologists, a collaborative study between the repository center and pathology panel for lymphoma clinical studies. *Journal of Clinical Oncology*, 3(1):25–38.

Müller, K.-R., Mika, S., Rätsch, G., Tsuda, K., and Schölkopf, B. (2001). An introduction to kernel-based learning algorithms. *IEEE TRANSACTIONS ON NEURAL NETWORKS*, 12(2):181.

Pantanowitz, L., Valenstein, P. N., Evans, A. J., Kaplan, K. J., Pfeifer, J. D., Wilbur, D. C., Collins, L. C., Colgan, T. J., et al. (2011). Review of the current state of whole slide imaging in pathology. *Journal of pathology informatics*, 2(1):36.

Peng, H., Long, F., and Ding, C. (2005). Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on pattern analysis and machine intelligence*, 27(8):1226–1238.

R Core Team (2016). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Rish, I. (2001). An empirical study of the naive bayes classifier. In *IJCAI 2001 workshop on empirical methods in artificial intelligence*, volume 3, pages 41–46. IBM New York.

Rokach, L. (2009). Ensemble methods in supervised learning. In *Data mining and knowledge discovery handbook*, pages 959–979. Springer.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088):533–536.

Saeys, Y., Inza, I., and Larrañaga, P. (2007). A review of feature selection techniques in bioinformatics. *bioinformatics*, 23(19):2507–2517.

Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of research and development*, 3(3):210–229.

Sertel, O. (2010). *Image analysis for computer-aided histopathology*. PhD thesis, The Ohio State University.

Sleeman, J. P. and Thiele, W. (2009). Tumor metastasis and the lymphatic vasculature. *International Journal of Cancer*, 125(12):2747–2756.

Stenkvist, B., Bengtsson, E., Eriksson, O., Jarkrans, T., Nordin, B., and Westman-Naeser, S. (1983). Histopathological systems of breast cancer classification: reproducibility and clinical significance. *Journal of clinical pathology*, 36(4):392–398.

Sutha, K. and Tamilselvi, J. J. (2015). A review of feature selection algorithms for data mining techniques. *International Journal on Computer Science and Engineering*, 7(6):63.

Tafreshi, N. K., Gillies, R. J., and Morse, D. L. (2012). Molecular imaging of breast cancer lymph node metastasis. *European journal of radiology*, 81(0 1):S160.

Takada, M., Sugimoto, M., Naito, Y., Moon, H.-G., Han, W., Noh, D.-Y., Kondo, M., Kuroi, K., Sasano, H., Inamoto, T., et al. (2012). Prediction of axillary lymph node metastasis in primary breast cancer patients using a decision tree-based model. *BMC medical informatics and decision making*, 12(1):1.

Tang, J., Alelyani, S., and Liu, H. (2014). Feature selection for classification: A review. *Data Classification: Algorithms and Applications*, page 37.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288.

Urrea, V. and Calle, M. (2012). *AUCRF: Variable Selection with Random Forest and the Area Under the Curve*. R package version 1.1.

Valkonen, M., Kartasalo, K., Liimatainen, K., Nykter, M., Latonen, L., and Ruusuvuori, P. (2017). Metastasis detection from whole slide images using local features and random forests. *Cytometry Part A*.

Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory*. Springer-Verlag New York, Inc., New York, NY, USA.

Weaver, D. L., Krag, D. N., Manna, E. A., Ashikaga, T., Harlow, S. P., and Bauer, K. D. (2003). Comparison of pathologist-detected and automated computer-assisted image analysis detected sentinel lymph node micrometastases in breast cancer. *Modern pathology*, 16(11):1159–1163.

Witten, I. H. and Frank, E. (2011). *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

Wolpert, D. H. (1992). Stacked generalization. *Neural networks*, 5(2):241–259.

Wu, J.-L., Tseng, H.-S., Yang, L.-H., Wu, H.-K., Kuo, S.-J., Chen, S.-T., and Chen, D.-R. (2014). Prediction of axillary lymph node metastases in breast cancer patients based on pathologic information of the primary tumor. *Medical science monitor: international medical journal of experimental and clinical research*, 20:577.

Yu, L. and Liu, H. (2003). Feature selection for high-dimensional data: A fast correlation-based filter solution. In *ICML*, volume 3, pages 856–863.

Zeng, Z., Zhang, H., Zhang, R., and Yin, C. (2015). A novel feature selection method considering feature interaction. *Pattern Recognition*, 48(8):2656–2666.

Appendices

Feature selection	Feature selected
BFS+ NB	LBP-8, HOG69, numberOfNuc, eLBP-8, eHOG34
AUCRF	numberOfNuc,Contrast,Homogeneity,Energy,LBP-2,LBP-3,LBP-6,LBP-10, LBP-7,LBP-8,SIFT-NroOfFrames, LBP-9,eLBP-8,eLBP-4,Kurtosis,eContrast, eSIFT-NroOfFrames, eHomogeneity,eLBP-6,eLBP-9,Skewness, eLBP-3,LBP-4,LBP-5, eLBP-2,eLBP-10,eSkewness ,eLBP-7,Correlation, MSER4, LBP-1, eEnergy,SIFT-ScaleMean, SIFTScaleStd,eLBP-1, nucDensity, eSIFT-ScaleMean, eCorrelation, eSIFT-ScaleStd
BFS+ KNN	LBP-4, LBP-6, numberOfNuc, eHomogeneity, eLBP-8, eLBP-9
BFS+ RF	Correlation, LBP-3, LBP-6, LBP-8, MSER4, numberOfNuc, eContrast, eEnergy ,eSkewness, eSIFT-NroOfFrames, eLBP-4, eLBP-10, eHOG77
BFS + SVM-RBF	Energy, SIFT-NroOfFrames, LBP-2, LBP-4, LBP-6, LBP-10, HOG40, HOG42, MSER1, MSER4 numberOfNuc, eCorrelation, eHomogeneity, eSkewness, eSIFT-NroOfFrames eLBP-2, eLBP-6, eHOG68, eHOG74
mRMR	eLBP-8, Contrast, eHOG12, SIFT-ScaleStd, eHOG74, eSIFT-ScaleStd, eCorrelation, meanNucDist ,LBP-2, MSER2, eMSER1, MSER3, eHOG32, eHOG4, HOG24,Correlation, HOG75, HOG60, stdNucDist, eMSER4, eMSER2 ,eContrast, eLBP-2, HOG25,SIFT-ScaleMean, HOG57, HOG22, eSIFT-ScaleMean, eHOG78, eKurtosis, MSER4,HOG2 ,HOG63, HOG8, Energy, HOG23, eMSER3, HOG76, LBP-3, HOG56

LASSO	Correlation, Energy , Skewness, SIFT-ScaleMean, SIFT-ScaleStd, SIFT-NroOfFrames, LBP-1, LBP-3, LBP-4, LBP-5, LBP-6 , LBP-7 , LBP-10, HOG1 , HOG2, HOG3 , HOG4, HOG5 , HOG6 , HOG7, HOG11 , HOG14 , HOG15, HOG17, HOG18, HOG19, HOG20 , HOG21, HOG22, HOG23, HOG24, HOG25, HOG26, HOG28, HOG29, HOG30, HOG31, HOG33, HOG34 , HOG35, HOG38, HOG39, HOG41, HOG42 , HOG44 , HOG45, HOG46, HOG47, HOG49, HOG52, HOG57, HOG58, HOG59, HOG60, HOG61, HOG62, HOG63, HOG65, HOG66, HOG67, HOG68, HOG69, HOG70, HOG71, HOG74, HOG77, HOG78, HOG80, MSER3, MSER4, maxNucDist, meanNucDist, stdNucDist, numberOfNuc, nucDensity, eContrast, eEnergy, eSkewness, eKurtosis, eSIFT-ScaleMean, eSIFT-ScaleStd, eSIFT-NroOfFrames, eLBP-1, eLBP-4 , eLBP-5, eLBP-6, eLBP-7, eLBP-8, eHOG1, eHOG2, eHOG3, eHOG5, eHOG6, eHOG8, eHOG9, eHOG11, eHOG13, eHOG4, eHOG15, eHOG16 , eHOG17, eHOG18, eHOG19, eHOG21, eHOG22, eHOG23, eHOG24, eHOG26 , eHOG28, eHOG29, eHOG31, eHOG35, eHOG40, eHOG45, eHOG46, eHOG47, eHOG48, eHOG49, eHOG51, eHOG54, eHOG55, eHOG56, eHOG58, eHOG59, eHOG60, eHOG62, eHOG63, eHOG64, eHOG66, eHOG67, eHOG68, eHOG69, eHOG72, eHOG73, eHOG74, eHOG75, eHOG76, eHOG77, eHOG78, eHOG81, eMSER1, eMSER2, eMSER3, eMSER4